Quantifying Near-Threshold CMOS Circuit Robustness

Sean Keller*, Siddharth S. Bhargav†, Chris Moore*, Alain J. Martin*

*Department of Computer Science
California Institute of Technology
Pasadena, CA 91125, USA
{sean,cc,alain}@async.caltech.edu

†Department of Electrical Engineering
University of Southern California
Los Angeles, CA 90089, USA
ssbharga@usc.edu

***Abstract*—In order to build energy efficient digital CMOS circuits, the supply voltage must be reduced to near-threshold. Problematically, due to random parameter variation, supply scaling reduces circuit robustness to noise. Moreover, the effects of parameter variation worsen as device dimensions diminish, further reducing robustness, and making parameter variation one of the most significant hurdles to continued CMOS scaling. This paper presents a new metric to quantify circuit robustness with respect to variation and noise along with an efficient method of calculation. The method relies on the statistical analysis of standard cells and memories resulting an an extremely compact representation of robustness data. With this metric and method of calculation, circuit robustness can be included alongside energy, delay, and area during circuit design and optimization.**

I. Introduction

It is difficult to design efficient and robust modern binary digital systems; the sheer complexity of utilizing upwards of a billion devices [1] necessitates the use of numerous levels of logical abstraction throughout the design flow. Errors introduced at different levels of abstraction can result in circuits that fail to function as expected for a number of reasons (*e.g.,* timing, design, and functional failures) [2]. Understanding and quantifying these different modes of failure is important, but failures in the *base digital assumption* supersede all other failures. If a gate cannot switch between logic values, then it cannot perform computation, and assuring correctness with respect to *e.g.,* timing, is moot. Functional failures of this sort can be further divided into many classes [3]; the focus of this paper is on active device parametric failures [4], *i.e.,* failures caused by one of the most significant hurdles for the future of CMOS scaling [5]: parameter variation.

Parameter variation is caused by stochastic process variation and intrinsic parameter fluctuations (IPF); it is the primary reason why modern digital circuits that function at the process nominal supply voltage (V_{DD}) eventually fail as the supply is lowered [6]. More importantly, parameter variation makes functional digital circuits less robust and hence less reliable [6]–[14]. This reduction in robustness may be of little consequence at the process nominal V_{DD}, but, as V_{DD} is lowered, it becomes a critical design concern. Problematically, in order to minimize the power consumption and energy demands of modern digital CMOS circuits, the supply voltage must be scaled sub-threshold or near-threshold [2], [8], [15]–[20]. As such, in order to build reliable low-power digital systems, it is essential to quantify circuit robustness as a function of parameter variation, which is the primary goal of this paper.

The prevailing trend is to perform a simple statistical analysis of worst-case gates and to choose a minimum V_{DD} above which most (or many) gates are likely to function despite parameter variation [6]. The problem with this type of analysis is that it may not be sufficient in real circuits due to the presence of electrical noise. Noise can be mitigated but is fundamentally unavoidable and has proven to be a limiting effect in engineering digital systems for decades [21]. This paper proposes a metric and method with which to quantify circuit robustness in terms of parameter variation with respect to noise. Moreover, the method presented is efficient and scalable. The computationally expensive component is limited to a small set of cells that make up modern standard cell libraries and memories, and the calculation of robustness cost is linear in the number of instances of these cells (typically in the range of millions to billions).

The remainder of this paper is organized as follows. Section II reviews background material on parameter variation and circuit noise analysis. Section III introduces the notion of circuit robustness and static noise margins. Section IV details the method for calculating robustness for inverters, and Section V extends the method to a larger set of CMOS gates. Section VI discusses related works, and finally, Section VII concludes the paper and discusses potential future research.

II. Background

A. Parameter Variation

In modern CMOS technologies, device parameters such as channel length, oxide thickness, dopant concentration, etc.

can have significant deviations from their nominal values due to process-induced and intrinsic parameter fluctuations [22]. Process variability can be considered a global, predictable, and gradual skew in device characteristics introduced by the complexity of manufacturing chips [23] (*e.g.,* from thermal gradients during fabrication [24]). Intrinsic parameter fluctuations are truly statistical in nature and cause significant deviations from device to device within a chip. Intrinsic variations can be attributed to atomistic effects (*e.g.,* random dopant fluctuation (RDF)) and device structure variations (*e.g.,* line edge roughness (LER)) [22], [23], [25]. There are a number of different ways to characterize and partition these effects, and the approach used in this paper is to consider a global component wherein all devices on a chip are affected in the same way, and a local component wherein each device on a chip has a number of statistical parameters drawn from distributions with mean values set by the global skew. This style of partitioning variation is not as accurate as a full combined statistical model, but it is a good, albeit slightly pessimistic approximation [23].

Considering variation in terms of a global and a local component simplifies statistical analysis and still permits the circuit designer to choose, for example, a worst-case 3σ global corner wherein the die that fall outside of this range are assumed not to yield and should not be optimized for. For circuits operating subthreshold, the local component of variation is dominated by RDF and is accurately modeled by normally distributed uncorrelated device threshold (V_t) variation [26]. Near-threshold, local variation does exhibit some degree of spatial correlation, and at the process-nominal V_{DD} spatial correlation is significant and cannot be ignored. This increase in the spatial correlation of local variation as a function of V_{DD} can be attributed to the fact that channel-length variation has little effect on devices operating subthreshold but becomes the dominant effect at approximately twice the threshold voltage [26]. Channel length variation is spatially correlated between devices within some radius, and is straightforward to model [23], [26], [27]. Given that the focus of this paper is to quantify the robustness of low-power subthreshold and near-threshold circuits, local parameter variation is treated as random and uncorrelated; however, the effects of spatial correlation can be included. Furthermore, SPICE simulations, along with foundry-provided statistically-extracted BSIM4 models, are used throughout this paper as a basis for correctness; these models are considered accurate over the entire device operating range [28].

B. Circuit Noise

Circuit noise can be partitioned into a physical component (*e.g.,* thermal noise) and a man-made digital switching component [21]. The dominant sources of physical noise in modern CMOS (which have significant impact on RF CMOS circuits) are $1/f$ noise and thermal noise [29]. Switching noise is caused by the rapid full-rail voltage swings typical in digital systems, and includes cross-talk (due to capacitive and inductive coupling), charge sharing, supply-rail and ground noise, and substrate noise. These switching-noise sources dominate physical noise by several orders of magnitude in digital circuits, and they must be accounted for in the design margins in order to build robust digital systems (even in the absence of appreciable parameter variation) [30]. Accurate modeling of each switching-noise source is possible, but highly impractical for the simulation and analysis of large circuits (millions or billions of devices). It is, however, possible to lump all switching-noise sources together into equivalent series voltage sources between gates [30]. These noise voltage sources are most accurately modeled as time-varying (*i.e.,* AC) sources [31], but using a static DC voltage is an acceptable approximation [21].

C. Static DC Analysis

Logic-gates in modern technologies exhibit a number of frequency-dependent effects, and incorporating these effects greatly increases the complexity of analysis. Fortunately, static DC analysis has proven to be an excellent basis for a wide range of digital circuit characterizations. The first works to discuss the requirements for functional digital circuits [32]–[34] exclusively perform DC analysis. Numerous modern works, *e.g.,* [14], [35], [36], also rely on the DC analysis of digital circuits, because in the context of determining functionality, noise resilience, and reliability, it is representative. Moreover, as discussed in Section I, timing failures (which probably cannot be quantified with DC analysis alone) fall outside of the scope of this work. In this paper static DC conditions are assumed throughout, and the corresponding canonical method of analysis, voltage transfer characteristics (VTCs)—the static output voltage of a gate as a function of input voltage—are used extensively.

III. Defining Circuit Robustness

Parameter variation and noise have a significant impact on circuit robustness, and the primary goal of this paper is to quantify this impact. To that end, it is necessary to define the notion of robustness with the intuition that increasing parameter variation tends to reduce robustness to noise. Consider two circuits, C_1 and C_2, operating at the same supply voltage; C_1 is more robust than C_2 if and only if C_1 can tolerate more noise than C_2. That is, as the circuit noise increases, C_2 fails to function before C_1. With statistical parameter variation, the notion of failure naturally becomes a probability. Robustness can be defined such that C_1 is more robust than C_2 if and only if for the same quantity of noise in both circuits the probability that C_1 fails is less than the probability that C_2 fails.

As discussed in Section I, the failures of interest are active device parametric failures, wherein a gate or memory erroneously changes state (between binary digital values) because of parameter variation. Circuit noise acts to make these failures more likely, and robust circuits need to function correctly despite parameter variation and switching noise. In order to quantify functional failures due to variation and noise it is necessary to define what it means for a gate or memory to change state. Toward this, consider the *base digital assumption*: the

abstraction of networks of transistors as logic-gates, and logic-gates as Boolean functions over Boolean logic-values. This abstraction relies on the definition of a mapping between logic-values and a physical quantity: the electrical potential of charge stored on capacitive gate nodes. In the simplest mapping, nodes near the supply rail potential, V_{DD}, represent a logic-1, and nodes near GND represent a logic-0; however, it is surprisingly difficult to define *near*. That is, it is difficult to give an exact (necessary and sufficient) mapping between node voltages and logic values for an arbitrary network of logic-gates, because each logic-gate *interprets* input voltages differently.

In a real CMOS circuit, no two gates are identical. They differ in function, topology, and sizing; and distinct instances of the same gate differ because of parameter variation. Consider an inverter; if a 0 is applied to its input, then a 1 is produced on its output. Similarly, a 1 at the input results in a 0 at the output. The problem is that it is possible—by way of intentional construction or parameter variation—to have two distinct inverters, INV_1 and INV_2, that behave differently. Suppose that for input voltages near V_{DD} or GND, INV_1 and INV_2 behave logically identically and correctly (*i.e.,* they invert), but for some input voltage, V_X, between V_{DD} and GND, INV_1 produces a 0 on its output and INV_2 produces a 1. In this situation, INV_1 and INV_2 *interpret* V_X differently. The situation is further complicated when the notion of the output voltage level is considered. That is, the output of INV_1 is really only a 0 when a subsequent gate *interprets* it as such, and so on down a chain of gates.

Since different gates have different *interpretations* of input voltages, the exact mapping between voltage levels and logic values needs to be defined in terms of this *interpretation* (as opposed to using a global bound). That is, suppose that worst-case boundaries on voltages are defined by V_H and V_L, where it is known that all gates in a circuit *interpret* voltages above V_H as a 1 and all voltages below V_L as 0; then the mapping of $V(G) > V_H \leftrightarrow 1$ and $V(G) < V_L \leftrightarrow 0$ is sufficient for some notion of correct operation, but it is not necessary. This distinction is important, because this sort of worst-case definition is simple but not practical for the analysis of modern low-voltage circuits.

Consider an example that demonstrates the trouble with using the worst-case definitions for V_H and V_L in low-voltage applications. Figure 1 depicts the VTCs for 100 instances of a minimum-size inverter in a modern 40-nm low-power bulk CMOS process with $V_{DD} = 200mV$; the curves vary significantly due to random parameter variation. These VTCs have remarkably similar shapes and are nearly identical modulo horizontal translation. As such, it is reasonable to consider defining $V_H = 180mV$ and $V_L = 20mV$ as worst-case output high and low voltages, respectively (these boundaries are also depicted by blue and red lines respectively in Figure 1). The problem with this worst-case output mapping is that the corresponding input voltages that yield a logical-1 on the output then range from $25mV$ to $150mV$; similarly, the input voltages that yield a logical-0 on the output range from $65mV$ to $195mV$. These ranges overlap, so a worst-case mapping of input voltages to logic values cannot be defined (the nonsensical worst-case mapping would be $V(G) > 65mV \leftrightarrow 1$ and $V(G) < 150mV \leftrightarrow 0$).

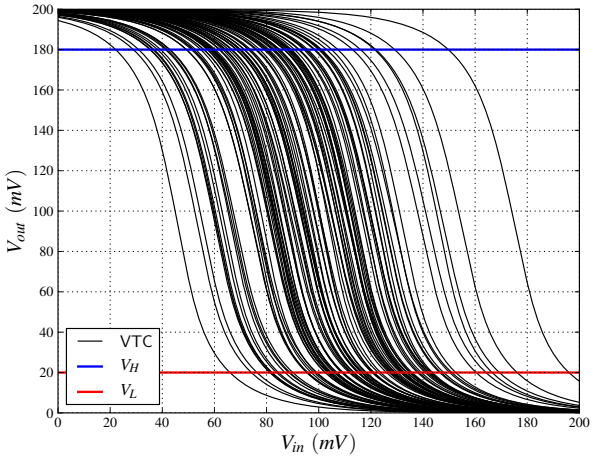

Fig. 1: Voltage transfer characteristics for 100 Monte Carlo trials of a minimum-size inverter in a commercial 40-nm low-power CMOS process utilizing foundry provided statistical models for local random parameter variation at the TT global corner ($V_{DD} = 200mV$ at $25°C$ TT-Corner).

A. Static Noise Margin

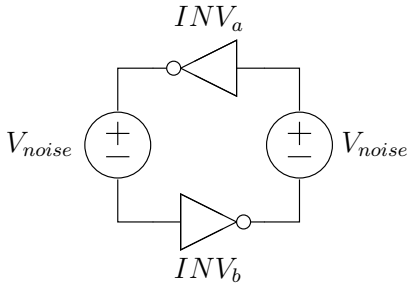

Fig. 2: Cross-coupled inverter pair and DC noise voltage sources.

A better approach to defining a local notion of *interpretation* stems from static noise margin (SNM) analysis. The static noise margin of cross-coupled inverters was first presented in [33], [34] and later clarified in [37] and [38]. Consider Figure 2; the SNM of this cross-coupled pair represents the largest DC noise voltage, V_{noise}, that can be applied between the bistable pair before the inverters switch state (between logic-0 and logic-1). If the SNM of a cross-coupled pair is less than or equal to zero (*e.g.,* due to parametric variation), then the pair is not bistable; *i.e.,* it is unable to hold two distinct logic states (a functional failure). If the SNM of the pair is infinitesimally greater than zero, then the cell can hold two distinct logic

states, but a diminutive noise can act to switch these states, so the cell is not robust. Given that noise is always present, all cross-coupled pairs of inverters in a digital system must have static noise margins in excess of the system noise in order to maintain state.[1]

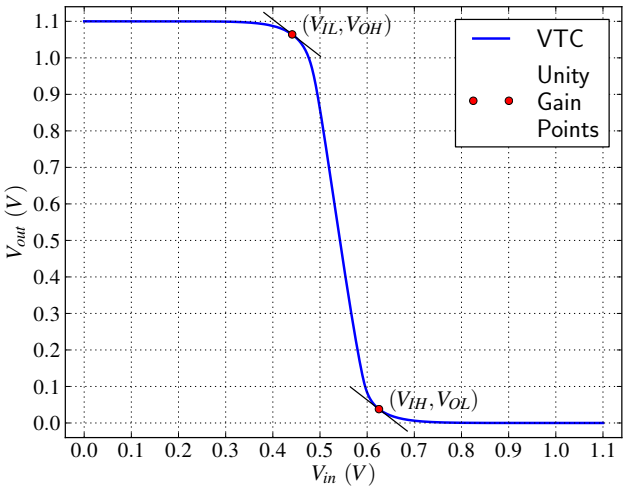

Fig. 3: Voltage transfer characteristic for a minimum-size inverter in a commercial 40-nm low-power CMOS process ($V_{DD} = 1.1V$ at $25°C$). The unity gain points are used to define the VTC parameters: $V_{OH}, V_{OL}, V_{IH}, V_{IL}$.

There are several mathematically equivalent methods used to measure static noise margins [37]. One such method involves analyzing the unity gain points ($|\frac{dV_{out}}{dV_{in}}| = 1$) of the voltage transfer characteristic. Consider INV_a (INV_b) from Figure 2: a static CMOS inverter consisting of a single NFET and PFET, with the VTC depicted in Figure 3. Both the functionality of the inverter and the definition of SNM rely on two properties of the VTC holding: (1) two unity gain points exist and (2) the slope between the unity gain points exceeds unity in absolute value [35]. From these unity gain points, four properties of an inverter VTC can be defined: $V_{OH}, V_{OL}, V_{IH}, V_{IL}$, as in Figure 3 (see [38] for details). (These four points are referred to as VTC parameters throughout.) The VTC parameters serve to demark definable boundaries between the voltages that are *interpreted* as a logic-1 or logic-0, and the undefined region of high-gain in between. That is, V_{IH} can be considered the lowest voltage that the inverter correctly *interprets* as a 1 and V_{IL} as the highest voltage that it correctly *interprets* as a 0. Similarly, V_{OH} can be considered the lowest voltage that the inverter will output as a 1, and V_{OL} the highest voltage that the inverter will output as a 0.

In general, when one gate *drives* another gate, a static noise margin can be defined. This static noise margin can be broken into two components: a noise margin high (NM_H) and a noise margin low (NM_L) (one for each logic value). Consider a pair of inverters, with INV_x driving INV_y. The two components of the corresponding noise margin are defined as

$$NM_H(INV_x, INV_y) = V_{OH}(INV_x) - V_{IH}(INV_y), \quad (1)$$

and,

$$NM_L(INV_x, INV_y) = V_{IL}(INV_y) - V_{OL}(INV_x). \quad (2)$$

The static noise margin is defined as the smaller of NM_H or NM_L.

$$SNM(INV_x, INV_y) = \\ min\left(NM_L(INV_x, INV_y), NM_H(INV_x, INV_y)\right). \quad (3)$$

These relations are implicit functions of V_{DD}.[2]

For cross-coupled inverters, as in Figure 2, INV_a drives INV_b, and INV_b drives INV_a, so two different static noise margins can be defined, $SNM(INV_a, INV_b)$ and $SNM(INV_b, INV_a)$. With a few assumption about the VTCs,[3] the condition that $SNM(INV_a, INV_b) > V_{noise} \cap SNM(INV_b, INV_a) > V_{noise}$ is a necessary and sufficient condition for differentiation of binary logic-values by way of the electrical potential stored on the output of each inverter [33], [34], [37]. The static noise margin of cross-coupled inverters plays an important role in quantifying circuit robustness, but the notion must be extended to incorporate parametric variability and generalized in order to apply it to arbitrary networks of gates.

B. Statistical Robustness

This section defines a robustness metric for cross-coupled inverters that includes parameter variation and noise by way of a statistical noise margin constraint. When considering two different circuits, C_1 and C_2, operating with the same supply voltage, C_1 is more robust than C_2 if and only if for the same quantity of noise in both circuits the probability that C_1 fails is less than the probability that C_2 fails. That is, for two different circuits C_1 and C_2,

$$ROB(C_1) > ROB(C_2) \leftrightarrow P(FAIL(C_1)) < P(FAIL(C_2)), \quad (4)$$

where ROB corresponds to circuit robustness and $FAIL$ to circuit failure.

Switching noise in digital circuits can be estimated with known-methods [21], [30], and, as with other common metrics, *e.g.,* power and cycle time, it can be reduced and optimized for (typically at some cost; *e.g.,* spreading wires reduces coupling noise at the expense of area). As such, the circuit designer can choose a noise margin target, NM_T: a minimum noise margin constraint for all gates.[4] If any gate has a noise margin less

[1] In real memories, *e.g.,* SRAM arrays, the SNM during both reading and writing of cells need to be considered [36]. Furthermore, ensuring a SNM of greater than zero is necessary, but it may not be sufficient for ensuring read stability and write-ability [11].

[2] Equations 1, 2, and 3 (and all dependent equations) are actually implicit functions of all operating parameters, *e.g.,* temperature, body potentials, etc.

[3] The VTCs must be monotonic and have a single inflection point.

[4] A unique noise margin target can be chosen for each gate (if desired). In this way, *noisy* gates can be assigned larger targets than *quiet* gates.

than or equal to the NM_T, then the gate is said to fail, as is the entire circuit containing the failing gate. Consider a cross-coupled inverter-pair, INV_a and INV_b, (as in Figure 2 with $V_{noise} = 0V$) operating at a particular V_{DD}. The probability of failure for a pair can then be defined such that

$$\begin{aligned} &P(FAIL(INV_a, NM_T) \cup FAIL(INV_b, NM_T)) \\ &= P(SNM(INV_a, INV_b) \leq NM_T \cup \\ &\quad SNM(INV_b, INV_a) \leq NM_T). \end{aligned} \tag{5}$$

For a circuit, C_a, consisting of n cross-coupled inverter-pairs, *i.e.,* $C_a = (INV_a^i, INV_b^i)$ for $i \in \{1, 2, ..., n\}$,

$$P(FAIL(C_a, NM_T)) = P\left(\bigcup_{i \in \{1,2,...,n\}} FAIL(INV_a^i, NM_T) \cup FAIL(INV_b^i, NM_T)\right) \tag{6}$$

These two relations treat both the probability of failure and SNM as random variables (RVs). In order to compute these quantities, the corresponding distributions and the effects of correlation are considered in Section IV. These two relations are generalized for application to arbitrary networks of gates in Section V.

IV. Calculating Robustness

One of the goals of this paper is to define a method for calculating robustness in such a way that it can be feasibly computed for large circuits (billions of gates), and which also fits in with the most prevalent method of system design, *i.e.,* standard-cell hierarchical digital circuit design. This necessitates the construction of a new compact model for statistical robustness with parameters that can be stored alongside timing and energy data in standard cell libraries. Moreover, the model must be defined such that the compact data is composable; *i.e.,* the robustness of an arbitrary network of standard cells must be computable by the composition of robustness data from member cells. In this way, the robustness of a large circuit (built out of standard cells) can be readily calculated.

A. Statistical VTC Parameters

Device parameter variation results in variation in the static noise margins of gates; the precise relationship depends on the type of parameter variation and the device operating regime (subthreshold see [14], [36], and above threshold see [39], [40]). The variation in SNM can be analyzed in terms of NM_H and NM_L variation (see Equation 3). Similarly, NM_H and NM_L can be considered in terms of the corresponding constituent VTC parameters, V_{OH}, V_{IH}, and V_{OL}, V_{IL}, respectively (see Equations 1 and 2). In modern bulk CMOS technologies, the output VTC parameters of a gate, V_{OH} and V_{OL}, can be considered regular (not random) variables.[5] The input VTC parameters, V_{IH} and V_{IL}, are normal random variables [36]. Consider Figure 1 (in Section III): for a particular gate (an inverter) operating at a particular supply voltage (200mV) the output VTC parameters, V_{OH} and V_{OL}, are nearly constant and close to V_{DD} and GND, respectively (consider the blue and red lines). The horizontal translation between this family of VTC curves—due to random parameter variation—corresponds to shifts in the input VTC parameters, V_{IH} and V_{IL}.

[5]First-order analysis in [14] finds V_{OH} and V_{OL} to be global constants dependent only on temperature when operating in the subthreshold regime. Including second order affects and near-threshold operation induces a dependence on V_{DD} and gate topology, so V_{OH} and V_{OL} are treated as regular variables.

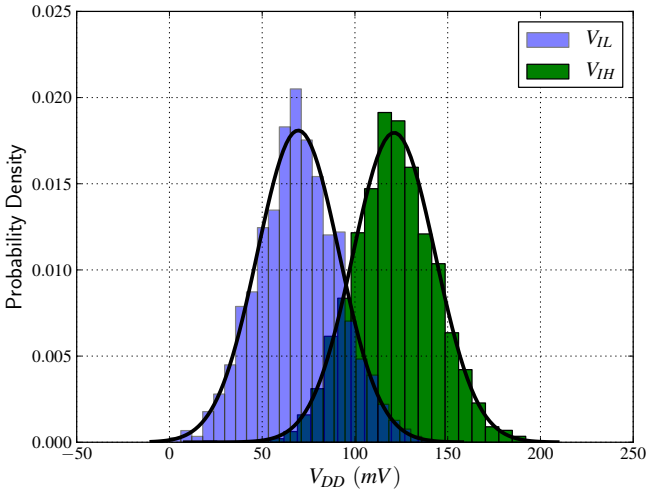

Fig. 4: V_{IH} and V_{IL} distributions for a minimum-size inverter in a commercial 40-nm low-power CMOS process at the TT-Corner ($V_{DD} = 200mV$ at $25°C$).

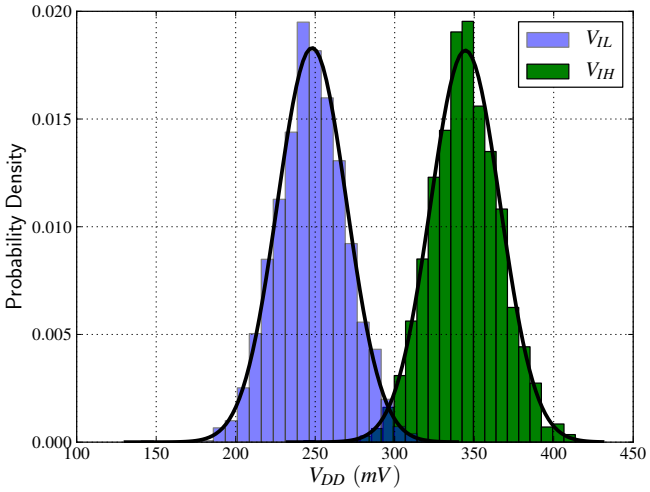

Fig. 5: V_{IH} and V_{IL} distributions for a minimum-size inverter in a commercial 40-nm low-power CMOS process at the TT-Corner ($V_{DD} = 600mV$ at $25°C$).

The input VTC parameter are normally distributed with mean and standard deviation determined by the supply voltage,

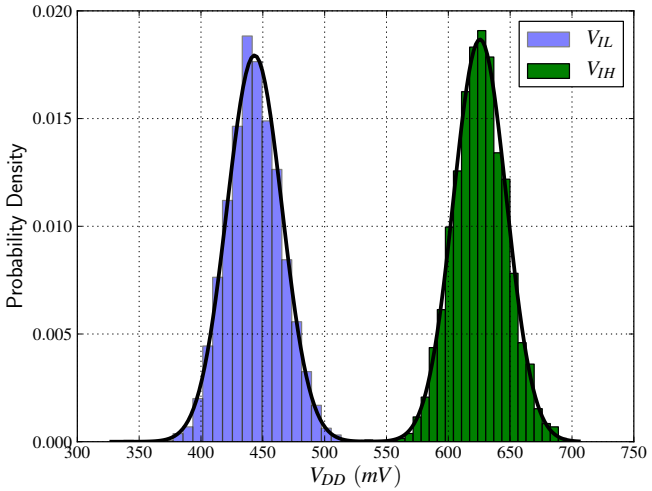

Fig. 6: V_{IH} and V_{IL} distributions for a minimum-size inverter in a commercial 40-nm low-power CMOS process at the TT-Corner ($V_{DD} = 1.1V$ at $25°C$).

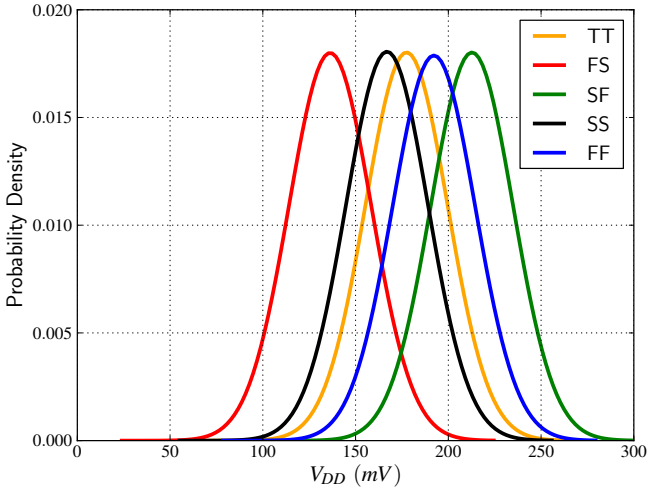

Fig. 7: V_{IH} distributions for a minimum-size inverter in a commercial 40-nm low-power CMOS process ($V_{DD} = 300mV$ at $25°C$). Global variation shifts the mean value for both V_{IH} and V_{IL}.

gate topology, temperature, and global corner. This is confirmed by the analysis of two standard cell libraries in different technologies and from different foundries (a 40-nm low-power process and a 65-nm low-power process). Both cell libraries contain hundreds of cells, and Anderson-Darling normality testing shows that neither V_{IH} nor V_{IL} have any significant

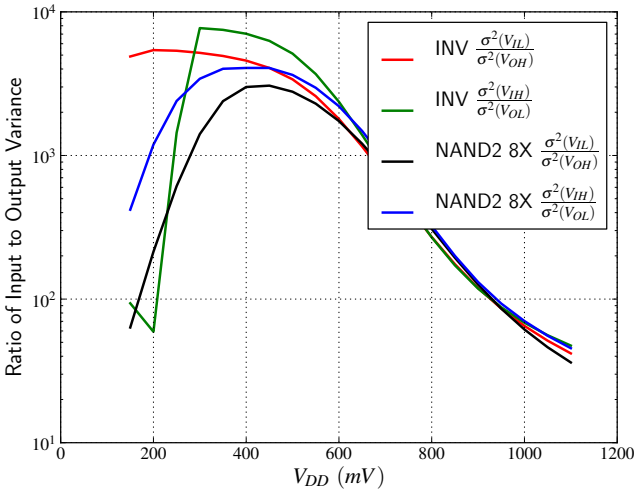

Fig. 8: Ratio of input VTC parameter variance to output VTC parameter variance in a commercial 40-nm low-power CMOS process ($25°C$, TT-Corner). The large ratio across the entire operating range makes it possible to approximate the output VTC parameters as regular variables, whereas the input VTC parameters are considered random variables.

departure from normality over the entire operating range.[6] Figures 4, 5, and 6 depict V_{IH} and V_{IL} histograms along with corresponding normal probability density functions (PDFs) for a minimum-size inverter operating sub-threshold, near-threshold, and at process nominal V_{DD}, respectively. Global variation simply skews the mean value, as depicted in Figure 7. Finally, Figure 8 further justifies the treatment of the output VTC parameters as regular variables: the spread of each input VTC parameter is several orders of magnitude greater than the corresponding output VTC parameter spread.

B. Statistical Noise Margins

At any particular global corner, local parameter variation is uncorrelated (see Section II-A), so the VTC parameters for distinct gates are independent. Consider two distinct inverters, INV_x driving INV_y; INV_x and INV_y have independent normally distributed input VTC parameters. From Equations 1 and 2 *and* the assumption that the corresponding output VTC parameters are regular variables, it follows that the corresponding NM_H and NM_L are also normally distributed RVs with mean and standard deviation given by

$$\mu(NM_H(INV_x, INV_y)) = V_{OH}(INV_x) - \mu(V_{IH}(INV_y)),$$
$$\sigma(NM_H(INV_x, INV_y)) = \sigma(V_{IH}(INV_y)), \quad (7)$$

[6]The nature of normality testing makes it difficult to make a stronger statement. Furthermore, it is extremely difficult to verify that the tails of purportedly normal distributions are actually normal; as such, treating V_{IH} and V_{IL} and normal RVs should be considered an approximation.

and

$$\mu(NM_L(INV_x, INV_y)) = \mu(V_{IL}(INV_y)) - V_{OL}(INV_x),$$
$$\sigma(NM_L(INV_x, INV_y)) = \sigma(V_{IL}(INV_y)), \quad (8)$$

where for any RV Z, $\mu(Z)$ and $\sigma(Z)$ denote the mean value and the standard deviation, respectively. Inconveniently, the statistical SNM does not follow directly from Equation 3 (due to the min function). If $NM_H(INV_x, INV_y)$ and $NM_L(INV_x, INV_y)$ are independent, order statistics can be used to directly calculate $SNM(INV_x, INV_y)$ [36]; however, they are not independent. From Figure 9, it is clear that the input VTC parameters are highly positively correlated, and it follows from this and Equations 7 *and* 8 that NM_H and NM_L are highly negatively correlated, which makes the direct calculation of SNM difficult. The approach taken in this paper is to use NM_H and NM_L directly to calculate the probability that a circuit fails, thus avoiding the need to compute SNM. In this way, the effects of correlation can be accounted for, and a general method for failure analysis is made possible.

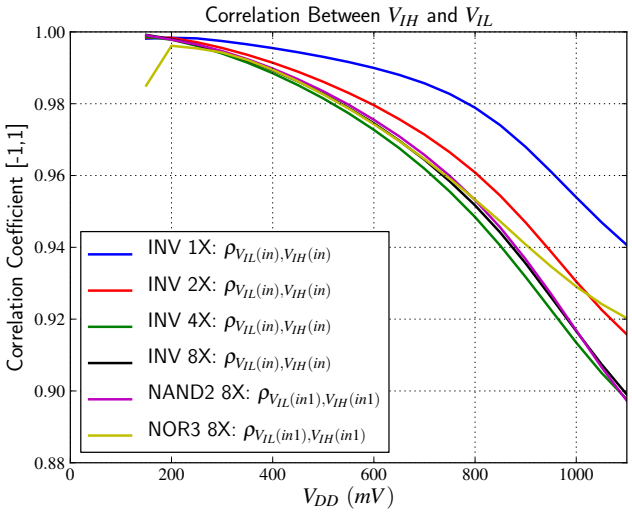

Fig. 9: Correlation between V_{IH} and V_{IL} in a commercial 40-nm low-power CMOS process ($25°C$, TT-Corner). These input VTC parameters are highly positively correlated across V_{DD} for a wide variety of gates.

C. Cross-coupled Inverter: Failure Probability

With a notion of statistical robustness (Section III-B) and statistical noise margins (Section IV-B) defined, it is possible to calculate the probability of cross-coupled inverter failure, and hence its robustness. Again, consider a cross-coupled inverter-pair, INV_a and INV_b, (as in Figure 2 with $V_{noise} = 0V$) operating at a particular V_{DD} and with a noise margin target of NM_T. Calculating the probability of failure (from Equation 5) necessitates the evaluation of $P(SNM(INV_a, INV_b) \leq NM_T \cup SNM(INV_b, INV_a) \leq NM_T)$. Assuming statistical independence, the disjunction can be treated as an addition, and Equation 5 reduces to

$$\begin{aligned} &P(FAIL(INV_a, NM_T) \cup FAIL(INV_b, NM_T)) \\ &= P(SNM(INV_a, INV_b) \leq NM_T) + \\ &\quad P(SNM(INV_b, INV_a) \leq NM_T). \end{aligned} \quad (9)$$

To calculate this quantity in closed-form, it is necessary to re-term this relation using NM_H and NM_L in lieu of SNM (as discussed in Section IV-B). In order to do this, upper and lower bounds on failure are determined, and then an approximation is given.

1) Upper Bound:

If $SNM(INV_a, INV_b) \leq NM_T$, then $NM_H(INV_a, INV_b) \leq NM_T$ and/or $NM_L(INV_a, INV_b) \leq NM_T$ (this follows directly from Equation 3). This can be stated in terms of probabilities as

$$\begin{aligned} &P(SNM(INV_a, INV_b) \leq NM_T) \\ &\leq P(NM_H(INV_a, INV_b) \leq NM_T \cup \\ &\quad NM_L(INV_a, INV_b) \leq NM_T). \end{aligned} \quad (10)$$

Due to the high degree of anti-correlation between NM_H and NM_L (see Section IV-B), the disjunction can be approximated as an addition and

$$\begin{aligned} &P(SNM(INV_a, INV_b) \leq NM_T) \\ &\leq P(NM_H(INV_a, INV_b) \leq NM_T) + \\ &\quad P(NM_L(INV_a, INV_b) \leq NM_T). \end{aligned} \quad (11)$$

Due to symmetry, a similar argument holds for $SNM(INV_b, INV_a)$, so combining Equation 9 and 11 yields an upper bound on the probability of failure for cross-coupled inverters. That is,

$$\begin{aligned} &P(FAIL(INV_a, NM_T) \cup FAIL(INV_b, NM_T)) \\ &\leq P(NM_H(INV_a, INV_b) \leq NM_T) + \\ &\quad P(NM_L(INV_a, INV_b) \leq NM_T) + \\ &\quad P(NM_H(INV_b, INV_a) \leq NM_T) + \\ &\quad P(NM_L(INV_b, INV_a) \leq NM_T). \end{aligned} \quad (12)$$

2) Lower Bound:

If $NM_H(INV_a, INV_b) \leq NM_T$ and $NM_L(INV_a, INV_b) \leq NM_T$, then $SNM(INV_a, INV_b) \leq NM_T$ (this follows directly from Equation 3). This can be stated in terms of probabilities as

$$\begin{aligned} &P(SNM(INV_a, INV_b) \leq NM_T) \\ &> P(NM_H(INV_a, INV_b) \leq NM_T \cap \\ &\quad NM_L(INV_a, INV_b) \leq NM_T). \end{aligned} \quad (13)$$

Due to the high degree of anti-correlation between NM_H and NM_L (see Section IV-B), the conditional probability of each event ($NM_H(INV_a, INV_b) \leq NM_T$, and $NM_L(INV_a, INV_b) \leq NM_T$) is less than the unconditional probability, so

$$\begin{aligned} &P(SNM(INV_a, INV_b) \leq NM_T) \\ &> P(NM_H(INV_a, INV_b) \leq NM_T) * \\ &\quad P(NM_L(INV_a, INV_b) \leq NM_T). \end{aligned} \quad (14)$$

Due to symmetry, a similar argument holds for $SNM(INV_b, INV_a)$, so combining Equation 9 and 14 yields a lower bound on the probability of failure for cross-coupled inverters. That is,

$$\begin{aligned} & P(FAIL(INV_a, NM_T) \cup FAIL(INV_b, NM_T)) \\ > & P(NM_H(INV_a, INV_b) \leq NM_T) * \\ & P(NM_L(INV_a, INV_b) \leq NM_T) + \\ & P(NM_H(INV_b, INV_a) \leq NM_T) * \\ & P(NM_L(INV_b, INV_a) \leq NM_T). \end{aligned} \quad (15)$$

3) Heuristic Approximation:

One way to approximate the probability of failure comes from the consideration of the cross-coupled pair as a whole. If INV_A is skewed such that it can barely *interpret* a logical-0 and INV_B is skewed such that is can barely *interpret* a logical-1 (or vice versa), then a failure is likely. That is, if $NM_H(INV_a, INV_b) \leq NM_T$ and $NM_L(INV_b, INV_a) \leq NM_T$, or if $NM_H(INV_b, INV_a) \leq NM_T$ and $NM_L(INV_a, INV_b) \leq NM_T$, then it is likely that $SNM(INV_a, INV_b) \leq NM_T$ or $SNM(INV_b, INV_a) \leq NM_T$. Empirically, with a small shift, δ, the lower bound *approximation* (given by Equation 15) leads to an accurate heuristic over a wide range of NM_T and V_{DD}. That is,

$$\begin{aligned} & P(FAIL(INV_a, NM_T) \cup FAIL(INV_b, NM_T)) \\ \approx & P(NM_H(INV_a, INV_b) \leq NM_T + \delta) * \\ & P(NM_L(INV_b, INV_a) \leq NM_T + \delta) + \\ & P(NM_H(INV_b, INV_a) \leq NM_T + \delta) * \\ & P(NM_L(INV_a, INV_b) \leq NM_T + \delta). \end{aligned} \quad (16)$$

D. Probability Computation

Finally, the Gauss error function, erf, and the cumulative distribution function (CDF) of the normal distribution can be used to compute the probability of failure. If Z is a normal random variable with mean μ and standard deviation σ, and c a constant, then

$$P(Z \leq c) = \frac{1}{2}\left(1 + erf(\frac{c-\mu}{\sigma\sqrt{2}})\right). \quad (17)$$

Consider an inverter INV_x driving another inverter INV_y, combining Equations 7, 8, and 17 yields

$$P(NM_H(INV_x, INV_y) \leq NM_T) = \frac{1}{2}\left[1 + erf\left(\frac{NM_T - (V_{OH}(INV_x) - \mu(V_{IH}(INV_y)))}{\sigma(V_{IH}(INV_y))\sqrt{2}}\right)\right]$$

and

$$P(NM_L(INV_x, INV_y) \leq NM_T) = \frac{1}{2}\left[1 + erf\left(\frac{NM_T - (\mu(V_{IL}(INV_y)) - V_{OL}(INV_x))}{\sigma(V_{IL}(INV_y))\sqrt{2}}\right)\right]. \quad (18)$$

Equation 18 can be applied directly to Equations 12, 15, and 16, thus yielding close-form equations for the probability of cross-coupled inverter failure. Note that the these expressions

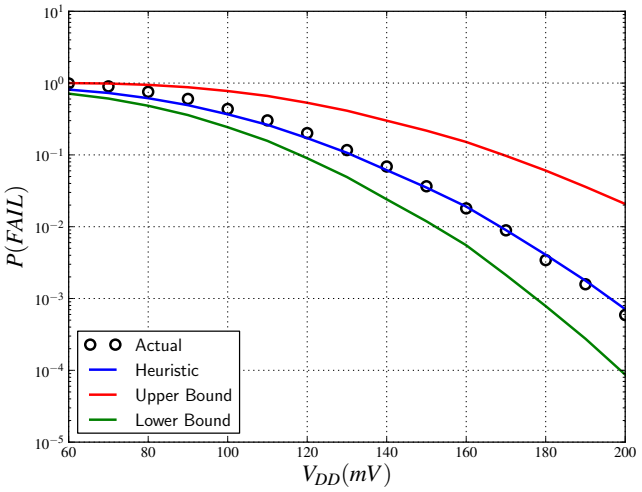

Fig. 10: Probability of minimum-size cross-coupled inverter-pair failure for $NM_T = 0mV$ in a commercial 40-nm low-power CMOS process ($25°C$, TT-Corner). For the heuristic approximation, the mean absolute error is 13%, and the maximum absolute error is 20% with $\delta = 4.2\%V_{DD}$.

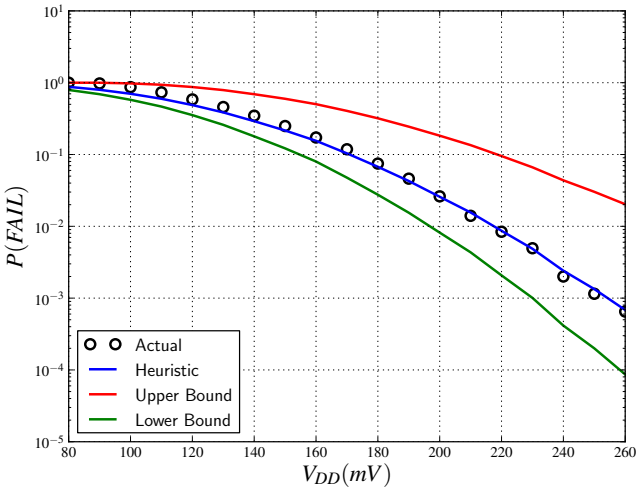

Fig. 11: Probability of minimum-size cross-coupled inverter-pair failure for $NM_T = 10\%V_{DD}$ in a commercial 40-nm low-power CMOS process ($25°C$, TT-Corner). For the heuristic approximation, the mean absolute error is 12%, and the maximum absolute error is 20% with $\delta = 3.2\%V_{DD}$.

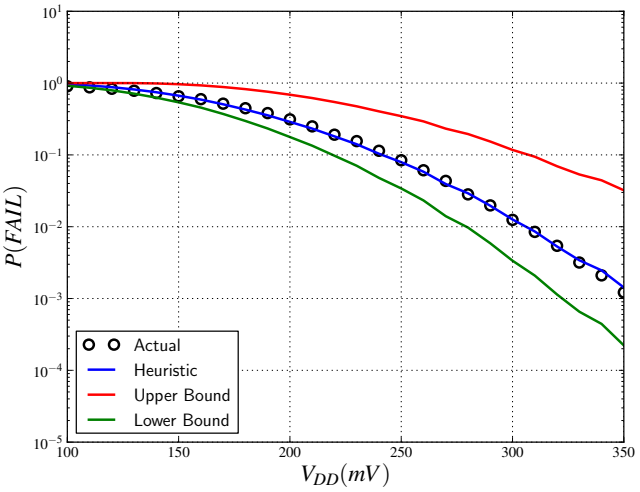

Fig. 12: Probability of minimum-size cross-coupled inverter-pair failure for $NM_T = 20\%V_{DD}$ in a commercial 40-nm low-power CMOS process ($25°C$, TT-Corner). For the heuristic approximation, the mean absolute error is 5.2%, and the maximum absolute error is 17% with $\delta = 2.2\%V_{DD}$.

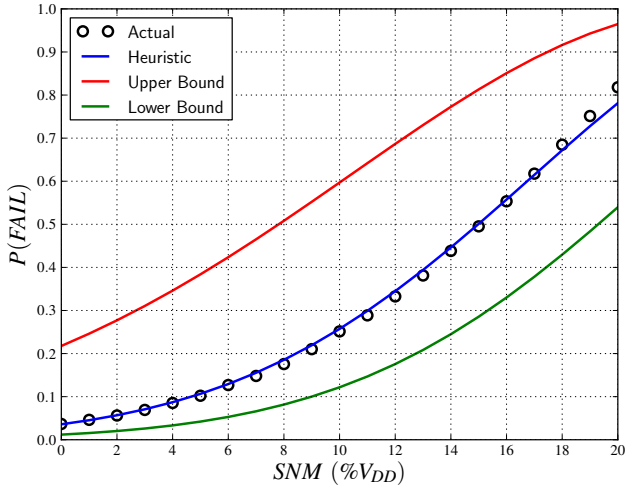

Fig. 13: Probability of minimum-size cross-coupled inverter-pair failure for $V_{DD} = 150mV$ in a commercial 40-nm low-power CMOS process ($25°C$, TT-Corner). For the heuristic approximation, the mean absolute error is 2.5%, and the maximum absolute error is 5.5% with $\delta = 4.3\%V_{DD}$.

for failure likelihood rely on an extremely compact set of real numbers:

- $V_{OH}(INV_{x,y})$
- $V_{OL}(INV_{x,y})$
- $\mu(V_{IH}(INV_{x,y}))$
- $\mu(V_{IL}(INV_{x,y}))$
- $\sigma(V_{IH}(INV_{x,y}))$
- $\sigma(V_{IL}(INV_{x,y}))$,

which is one of the goals of this section. That is, these are the only parameters needed in order to calculate the probability of failure, and hence robustness of a circuit. Figures 10, 11, and 12, plots the probability of failure for a crossed-coupled inverter pair against V_{DD}. Figure 13 plot the probability of cross-coupled inverter failure versus NM_T for a fixed supply voltage of $150mV$. Digital noise tends to be proportional to V_{DD} [41], so the NM_T is reported as a percentage of V_{DD}. Each of these figures depicts the upper bound, lower bound, and approximation for cross-coupled inverter failure probability, in addition to the actual (empirical) failure rate. Actual failures are calculated via Monte Carlo SPICE simulations with foundry provided statistical BSIM4 models.

Figures 10, 11, 12, and 13 also serve to exemplify why an accurate and simple closed-form approximation for the probability of failure is so important. In each of these plots, as V_{DD} increases linearly, the probability of failure decreases exponentially, and the size of the Monte Carlo simulations required to generate accurate failure rates increases exponentially. With a noise margin target of $10\%V_{DD}$ at $300mV$ the probability of failure is already less than 10^{-5}, so millions of Monte Carlo trials are necessary. A million such trials on modern computers with modern tools requires several core-hours of compute time. Furthermore, it is not uncommon for a modern microprocessor design to contain millions of cross-coupled inverters, so higher supply voltages with lower failure rates on the order of 10^{-9} or lower need to be considered. This corresponds to at least a four order of magnitude increase in compute time for a single temperature and V_{DD} of interest. To ensure reliability, multiple supply voltages and temperatures need to be considered, increasing the computation requirement by yet another order of magnitude. Optimization of transistor sizing can easily increase the computation requirement by another order of magnitude, resulting in a compute requirement in the realm of millions of core-hours. Finally, one of the goals of this paper is to extend this type of analysis to arbitrary gates (typical standard cell libraries contain hundreds of cells). This easily pushes the compute requirement to billions of core-hours. A closed-form approximation is more practical.

Finally, the probability of failure of a circuit C_a consisting of n cross-coupled inverter pairs ($C_a = (INV_a^i, INV_b^i)$ for $i \in \{1, 2, ..., n\}$) can easily be computed. Equation 6 can be re-written in terms of a global conjunction instead of disjunction

as

$$P(FAIL(C_a, NM_T)) = 1 - P\Big(\bigcap_{i\in\{1,2,\ldots,n\}} \neg\big(FAIL(INV_a^i, NM_T) \cup FAIL(INV_b^i, NM_T)\big)\Big). \quad (19)$$

Given the assumption of VTC parameter independence between gate pairs (see Section IV-B), the global conjunction can be treated as a product, giving a readily computable compact expression for the probability of failure and hence robustness of a circuit, one of the goals of this section. That is,

$$P(FAIL(C_a, NM_T)) = 1 - P\Big(\prod_{i\in\{1,2,\ldots,n\}} \neg\big(FAIL(INV_a^i, NM_T) \cup FAIL(INV_b^i, NM_T)\big)\Big). \quad (20)$$

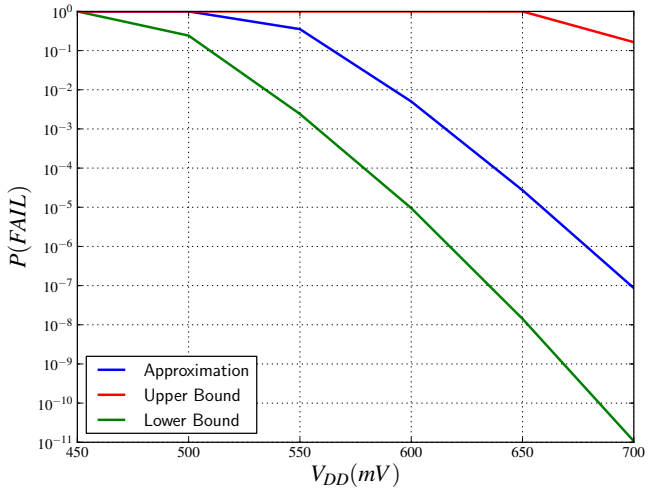

Fig. 14: Probability of failure for 2e28 minimum-size cross-coupled inverter-pairs with $NM_T = 20\%V_{DD}$ in a commercial 40-nm low-power CMOS process ($25°C$, TT-Corner, and $\delta = 2.2\%V_{DD}$).

With Equation 20, it is possible to quantify the probability of failure for an entire memory. Figure 14 gives the probability of failure for 2e28 independent cross-coupled inverter pairs (*i.e.,* a 32MB memory). Simulation with statistical SPICE is completely infeasible, so δ is taken from the considerably smaller experiments depicted in Figure 12.

E. Chains of Inverters: Failure Probability

The goal of this section is to extend the notions of noise margins and circuit robustness to arbitrary networks of inverters. This is necessary because the probability of failure of a linear chain of n inverters differs significantly from that of n cross-coupled inverters. At first glance, this seems counter-intuitive; several works (*e.g.,* [37]) have demonstrated that a cross-coupled pair of identical inverters can be modeled as—and is mathematically equivalent to—an infinite chain of identical inverters. Moreover, alternating worst-case (demonic) noise sources between a cross-coupled pair can be modeled as demonic alternating noise in an infinite chain, as depicted in Figure 15. The main idea behind this equivalence is that an infinite chain can be viewed as the unrolling of the *loop* that is a cross-coupled pair. When a bistable cross-coupled pair in steady state is perturbed by some voltage δV, the bistable pair either changes digital state, or the inverters act as a restorative filter, successively removing the δV disturbance one *iteration* at a time in the same exact way that a chain of inverters filters a δV disturbance. When the inverters are not identical, the two circuits no longer behave in the same way, and equivalence is lost. The intuition behind why they differ comes from further analysis of the heuristic approximation presented in Section IV-C.

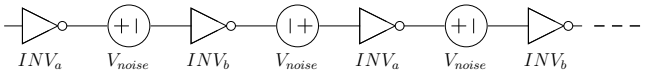

Fig. 15: Infinite chain construct: equivalent to the cross-coupled pair depicted in Figure 2.

Consider a cross-coupled inverter pair (INV_a, INV_b), where INV_a and INV_b behave differently due to parameter variation. With the infinite chain construct, this pair can be modeled as a never-ending alternating linear chain of INV_a driving INV_b driving INV_a driving INV_b, etc. (see Figure 15). Suppose that this inverter pair is not robust, *i.e.,* the static noise margin is just slightly larger than $0mV$ due to INV_a being skewed such that it can barely *interpret* a logical-0 and INV_b being skewed such that it can barely *interpret* a logical-1. Consider the state where the input of INV_a is a logical-0 and its output (the input of INV_b) is a logical-1. A small DC noise can raise the input voltage, thus causing INV_a to no longer *interpret* its input as a logical-0, thus resulting in a lowering of its output node voltage. This, in turn, can result in INV_b no longer *interpreting* its input as a logical-1, thus resulting in INV_b raising it's output node voltage. This, in turn, pushes INV_a even further away from interpreting its input as a logical-0, and so on down the *infinite chain* until the bistable pair 'flips' digital state.

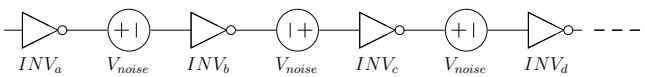

Fig. 16: Chain of inverters.

On the other hand, consider an actual linear chain of inverters, as in Figure 16. Due to parameter variation, each inverter in the chain behaves differently. Suppose that the chain begins with the identical sequence of skewed INV_a driving a skewed INV_b, but INV_b now drives a different

inverter INV_c. Again, consider the same state and event where the input of INV_a is a logical-0 and its output (the input of INV_b) is a logical-1, and a small DC noise raises the input voltage, thus causing INV_a to no longer *interpret* its input as a logical-0, resulting in a lowering of its output node voltage. This, in turn, results in INV_b no longer *interpreting* its input as a logical-1, resulting in INV_b raising it's output node voltage. Suppose, however, that INV_c is a robust inverter and completely restores the rather poor logical-0 generated by INV_B to approximately $0mV$. That is, the condition that causes INV_a and INV_b to flip state if configured as a cross-coupled pair may not cause a failure with INV_a and INV_b in a linear chain.

In order to calculate the probability of failure for a chain of inverters, the notion of what it means for a chain to fail must be defined. Not unexpectedly, this definition quickly becomes a problem of logic level *interpretation*, and a clear definition for what it actually means for a chain to fail does not immediately follow, but it is possible to sidestep the problem. That is, cross-coupled inverter static noise margin analysis avoids the definition of failure of an individual inverter by considering a bistable *loop*. Analogously, consider a chain of an even number, $n > 2$, of inverters. If the output of the last inverter in the chain is connected to the input of the first inverter, then the chain becomes a state-holding ring (*loop*). The definition of failure naturally follows as a failure of the ring to maintain state. Informally, the requirement of an even number of stages does not result in a loss of generality, as it is always possible to calculate a tight upper and lower bound on failure rate by considering a chain with one extra and one fewer inverters respectively.

As with the cross coupled inverter analysis, the NM_H and NM_L can be used to generate an upper bound, a lower bound, and an approximation for the probability of failure for chains of inverters. For a chain of inverters, the worst-case, demonic, DC noise consists of alternating positive and negative voltage sources acting contrary to the desired state of each inverter input. That is, if the desired input to a gate is logical-1, *i.e.,* V_{DD}, then a voltage source that acts to lower this electrical potential is said to act contrary to the desired state. In steady-state, a linear chain of n functional inverters consists of alternating sequences of 0 and 1 at the input of each inverter. As such, there are two possible digital states for such a chain: the sequence either begins with a 1 or it begins with a 0. Correspondingly, there are two states for alternating demonic noise sources; the first DC noise source is either positive or it is negative.

Consider a linear chain, CH_a, of n inverters with demonic noise sources, as in Figure 16 (with the constraint that n is an even integer greater than 2). The chain is said to fail if the corresponding ring, created by connecting the output of the last inverter to the input of the first inverter, fails to maintain state when all inverter inputs are properly initialized with alternating values of 0 and 1. The chain and ring fail with respect to a noise margin target, NM_T, when the ring fails to maintain state with demonic noise sources with $V_{noise} = NM_T$ or $V_{noise} = -NM_T$. Given the assumption of statistical independence between the noise margins of different gates pairs (as discussed in Section IV-B), the probability of chain failure can be analyzed and computed in terms of the NM_H and NM_L of pairs of gates.

1) Heuristic Upper Bound:

Consider a labeling of inverters in the chain CH_a such that the first inverter is labeled as INV_1, the second inverter as INV_2, and so on with the last inverter being INV_n. If the chain fails, then it follows that there exists some inverter pair in the chain, INV_i driving INV_{i+1}, with $NM_H(INV_i, INV_{i+1}) \leq NM_T$ and/or $NM_L(INV_i, INV_{i+1}) \leq NM_T$, which leads to the same probabilistic upper bound for cross-coupled pairs which was discussed in Section IV-C1. For chains, however, this is not a tight upper bound. Empirically, the cross-coupled pair heuristic approximation (see Section IV-C3) leads to a tighter upper bound for chains of gates. Consider two connected pairs of inverters, the set (INV_i, INV_{i+1}, INV_{i+2}); if the chain fails, then it is likely that either (1) $NM_L(INV_i, INV_{i+1}) \leq NM_T)$ and $NM_H(INV_{i+1}, INV_{i+2}) \leq NM_T)$, and/or (2) $NM_H(INV_i, INV_{i+1}) \leq NM_T)$ and $NM_L(INV_{i+1}, INV_{i+2}) \leq NM_T)$. With the assumption of statistical independence, the upper bound on the probability of failure for the chain can be approximated as,

$$
\begin{aligned}
& P(FAIL(CH_a, NM_T) \\
\lesssim\ & P\Bigg(\bigcup_{i \in \{1,2,\ldots,n-2\}} \\
& \big((NM_H(INV_i, INV_{i+1}) \leq NM_T + \delta u \cap \\
& \quad NM_L(INV_{i+1}, INV_{i+2}) \leq NM_T + \delta u) \\
& \cup (NM_L(INV_i, INV_{i+1}) \leq NM_T + \delta u \cap \\
& \quad NM_H(INV_{i+1}, INV_{i+2}) \leq NM_T + \delta u) \big) \Bigg) \qquad (21)
\end{aligned}
$$

where δu is a small constant used to maintain the boundary over a wide range of NM_T and V_{DD}. The directly computable form of this follows directly from Section IV-A.

2) Lower Bound: A heuristic for the lower bound has the same form, but a small constant δl must be subtracted from the NM_T.

$$
\begin{aligned}
& P(FAIL(CH_a, NM_T) \\
\gtrsim\ & P\Bigg(\bigcup_{i \in \{1,2,\ldots,n-2\}} \\
& \big((NM_H(INV_i, INV_{i+1}) \leq NM_T - \delta l \cap \\
& \quad NM_L(INV_{i+1}, INV_{i+2}) \leq NM_T - \delta l) \\
& \cup (NM_L(INV_i, INV_{i+1}) \leq NM_T - \delta l \cap \\
& \quad NM_H(INV_{i+1}, INV_{i+2}) \leq NM_T - \delta l) \big) \Bigg). \qquad (22)
\end{aligned}
$$

3) Approximation: As expected, the heuristic approximation follows from the upper and lower bound heuristics.

$$
\begin{aligned}
&P(FAIL(CH_a, NM_T) \\
&\approx P\Bigg(\bigcup_{i\in\{1,2,\ldots,n-2\}} \\
&\quad \big((NM_H(INV_i, INV_{i+1}) \leq NM_T + \delta \cap \\
&\quad\ NM_L(INV_{i+1}, INV_{i+2}) \leq NM_T + \delta) \\
&\quad \cup (NM_L(INV_i, INV_{i+1}) \leq NM_T + \delta \cap \\
&\quad\ NM_H(INV_{i+1}, INV_{i+2}) \leq NM_T + \delta)\big)\Bigg). \qquad (23)
\end{aligned}
$$

Empirically, δu and δl can be defined in terms of δ. For the devices considered in this paper: INV, NAND2, NOR3, NAND3, NOR3, AOI21, and for noise margin targets between $0\%V_{DD}$ and $20\%V_{DD}$, a relative offset of $3\%V_{DD}$ is sufficient. That is, $\delta u = \delta l = \delta + 3\%V_{DD}$. Figures 17, 18, and 19 depict the upper bound, lower bound, and approximations for a chain of 20 inverters.

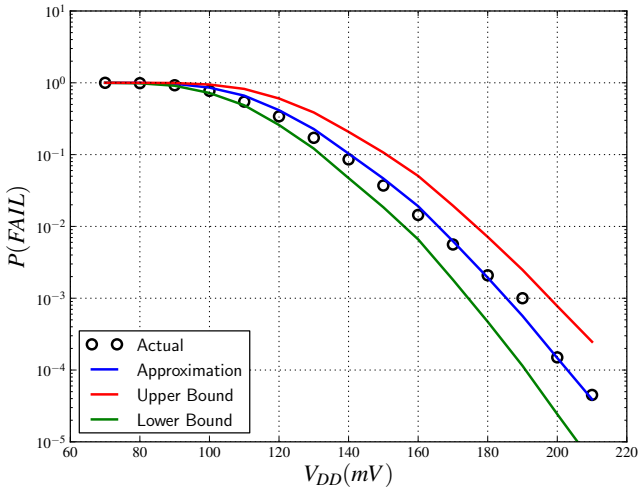

Fig. 17: Probability of chain of 20 inverters failing with $NM_T = 0\%V_{DD}$ in a commercial 40-nm low-power CMOS process ($25°C$, TT-Corner). For the heuristic approximation, the mean absolute error is 17%, and the maximum absolute error is 43% with $\delta = -3.2\%V_{DD}$.

Finally, a circuit, C_a, composed of n chains of inverters, is said to fail if any chain fails. That is, with chain labeled as $CH_1, CH_2, ..., CH_n$,

$$
\begin{aligned}
&P(FAIL(C_a, NM_T) = \\
&P\left(\bigcup_{i\in\{1,2,\ldots,n\}} FAIL(CH_i, NM_T) \right). \qquad (24)
\end{aligned}
$$

As with the cross-coupled inverter analysis in Section IV-D, Equation 24 can be re-written in terms of a global conjunction

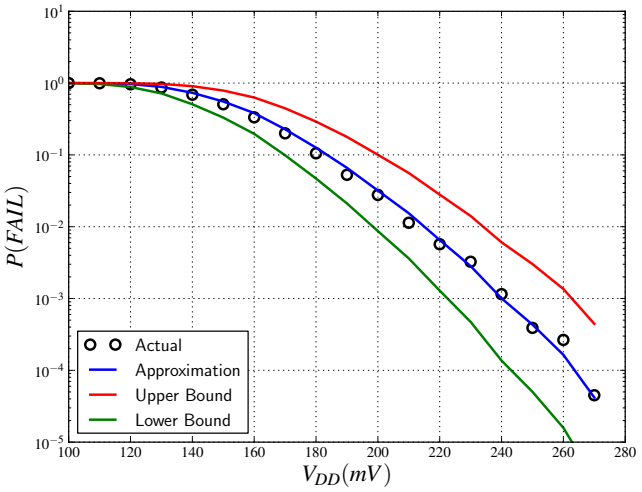

Fig. 18: Probability of chain of 20 inverters failing with $NM_T = 10\%V_{DD}$ in a commercial 40-nm low-power CMOS process ($25°C$, TT-Corner). For the heuristic approximation, the mean absolute error is 13%, and the maximum absolute error is 38% with $\delta = -2.3\%V_{DD}$.

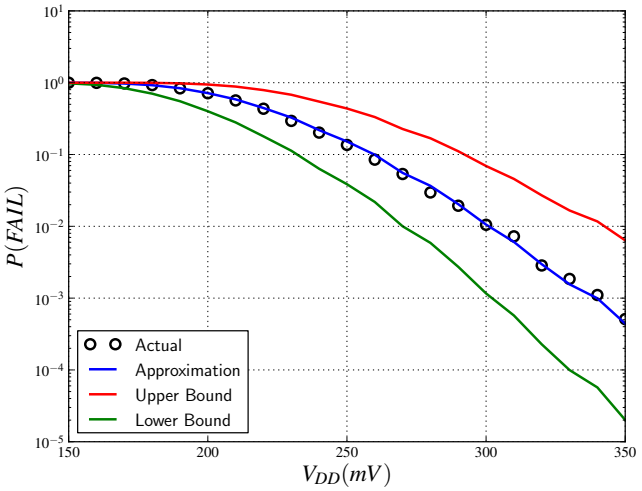

Fig. 19: Probability of chain of 20 inverters failing with $NM_T = 20\%V_{DD}$ in a commercial 40-nm low-power CMOS process ($25°C$, TT-Corner). For the heuristic approximation, the mean absolute error is 6.8%, and the maximum absolute error is 24% with $\delta = -1.8\%V_{DD}$.

instead of disjunction as

$$P(FAIL(C_a, NM_T)) = 1 - P\left(\bigcap_{i\in\{1,2,\ldots,n\}} \neg FAIL(CH_i, NM_T)\right). \quad (25)$$

Given the assumption of VTC parameter independence between gate pairs, and hence chains (see Section IV-B), the global conjunction can be treated as a product, giving a readily computable compact expression for the probability of failure and hence robustness of a circuit consisting of chains of inverters, one of the goals of this section. That is,

$$P(FAIL(C_a, NM_T)) = 1 - P\left(\prod_{i\in\{1,2,\ldots,n\}} \neg FAIL(CH_i, NM_T)\right). \quad (26)$$

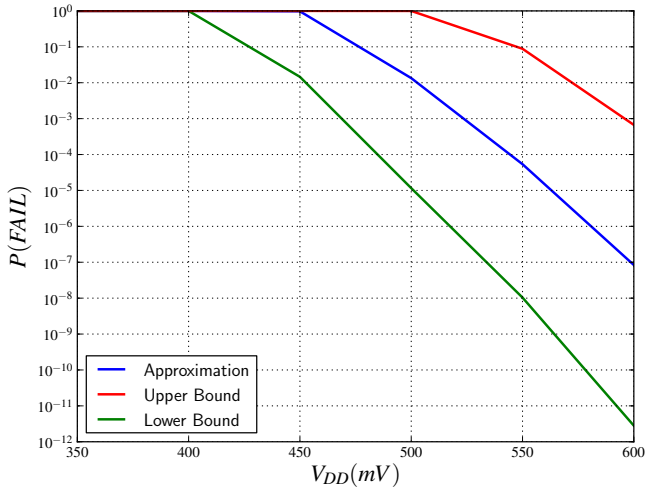

Fig. 20: Probability of failure for 2*2e28 minimum-size inverters in chains with $NM_T = 20\%V_{DD}$ in a commercial 40-nm low-power CMOS process ($25°C$, TT-Corner, and $\delta = -1.8\%V_{DD}$).

Using Equation 26, Figure 20 gives the probability of failure for 2*2e28 independent inverters in the form of chains. As with the cross-coupled pairs, simulation with statistical SPICE is infeasible, so δ is taken from the considerably smaller experiments depicted in Figure 19. The probability of failure of chains of inverters is considerably lower than that of cross-coupled pairs (with the same number of devices, noise-margin target, and V_{DD}), as depicted in Figure 21. The failure probabilities are similar for cross-coupled pairs of inverters operating $50-100mV$ above the inverter chain supply voltage. This is the first work to quantify and compare these two very different devices' configurations and corresponding probabilities of failure.

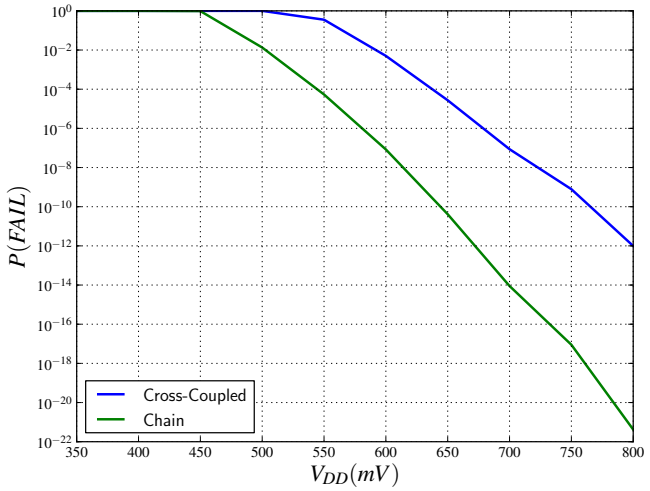

Fig. 21: Probability of failure for 2*2e28 minimum-size inverters in chains compared to that of 2e28 minimum size cross-coupled pairs with $NM_T = 20\%V_{DD}$ in a commercial 40-nm low-power CMOS process ($25°C$, TT-Corner).

V. Generalized Circuit Robustness

The goal of this section is to extend the notions of static noise margins to a larger gate set than that of inverters alone. As with the analysis in Section IV, the main goal is to generate a composable robustness metric, so that the robustness of an arbitrary network of standard cells can be easily computed.

A. VTC Parameters of Combinational Gates

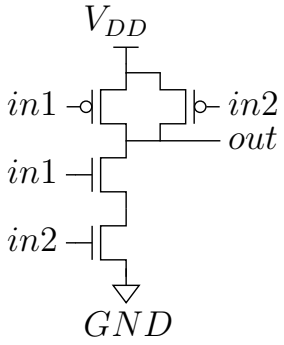

Fig. 22: NAND2.

Consider a combinational CMOS 2-input NAND gate, NAND2, with nodes labeled as in Figure 22. If the input nodes, $in1$ and $in2$, are treated independently, then the VTC describes V_{out} as a function of both V_{in1} and V_{in2}, as depicted in Figures 23 and 24. Figure 23 provides a three dimensional view and, Figure 24 plots the VTC in the $V_{in1} \times V_{in2}$ plane with V_{out} encoded by color. The partial derivatives $\frac{\partial V_{out}}{\partial V_{in1}}$ and $\frac{\partial V_{out}}{\partial V_{in2}}$ describe two continuums of unity gain points depicted

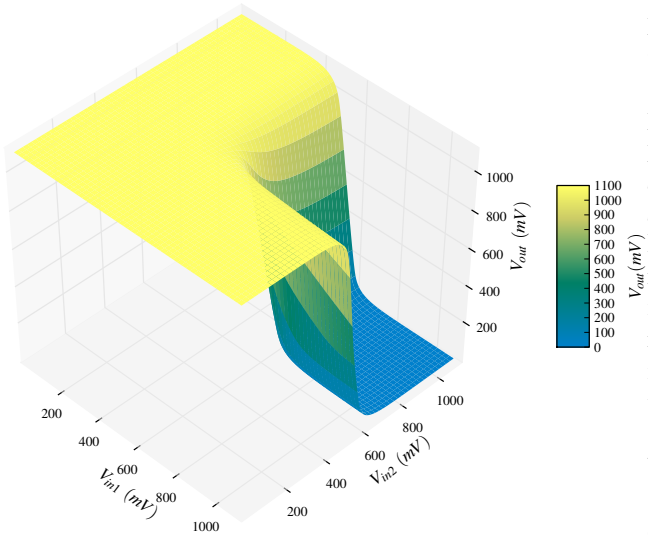

Fig. 23: Voltage transfer characteristic for the minimum-size NAND2 (depicted in Figure 22) in a commercial 40-nm low-power CMOS process ($V_{DD} = 1.1V$, $25°C$, TT-Corner)

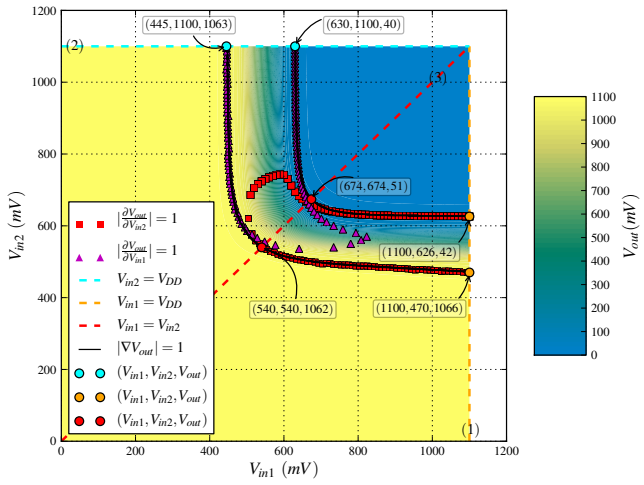

Fig. 24: Voltage transfer characteristic for the minimum-size NAND2 (depicted in Figure 22) in a commercial 40-nm low-power CMOS process ($V_{DD} = 1.1V$, $25°C$, TT-Corner).

by purple triangles and red squares, respectively, in Figure 24. The gradient is given by

$$\nabla V_{out} = \frac{\partial V_{out}}{\partial V_{in1}}\mathbf{i} + \frac{\partial V_{out}}{\partial V_{in2}}\mathbf{j}, \tag{27}$$

where $\mathbf{i}$ and $\mathbf{j}$ are the unit vectors in the $V_{in1} \times V_{in2}$ plane. The two continuums of unity gain points defined by $|\nabla V_{out}| = 1$ (depicted by a black line in Figure 24) are analogous to an inverter's two unity gain points given by $|\frac{dV_{out}}{dV_{in}}| = 1$ in Section III-A. In fact, for an inverter the two measures are identical, *i.e.*, $|\nabla V_{out}| \equiv |\frac{dV_{out}}{dV_{in}}|$. Moreover, the magnitude of ∇V_{out} is probably the most general measure for determining unity gain points, as it is applicable to any gate regardless of the number of inputs.

For noise margin analysis, choosing individual unity gain points as representative approximations is an important simplification, and individual points can be chosen by considering *slices* of the VTC (planes orthogonal to $V_{in1} \times V_{in2}$). Three *slices* of the NAND2 VTC are depicted by dashed lines in Figure 24. These three *slices* are of particular interest for two reasons. First, they give the upper and lower unity-gain bounds in terms of V_{in1} and V_{in2}. Second, they correspond to a logical reduction of the NAND2 to that of an inverter. That is, if either input is tied to logical-1 or if both inputs are tied together, then the NAND2 is functionally equivalent to an inverter. In Figure 24 the three possible inverter-equivalent slices are depicted by: (1) an orange dashed-line corresponding to tying $in1$ to V_{DD} and sweeping $in2$ from GND to V_{DD}, (2) a light-blue dashed-line corresponding to tying $in2$ to V_{DD} and sweeping $in1$ from GND to V_{DD}, and (3) a red dashed-line generated by tying $in1$ to $in2$ and sweeping them together.

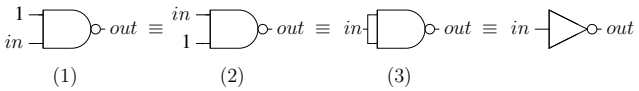

Fig. 25: NAND2 inverter equivalence.

The idea of inverter-equivalence is general and can be used to generate the boundary unity-gain points for arbitrary gates. Consider an inverting binary CMOS gate, G, with n inputs and a single output. Gate G can be made to act logically as a single input/output inverter for some assignment of inputs where inputs can be tied together, tied to `1`, or tied to `0`. Since G is an inverting CMOS gate, one or more inverter-equivalent input assignments exist, and the assignments depend on the topology of G. The three inverter-equivalent slices from Figure 24 are depicted at the gate level in Figure 25.

A general notion of an inverter equivalent assignment is helpful. Let G be an inverting binary CMOS gate with k inputs labeled as $in1, in2, ..., ink$, a single output, out, and with functionality defined by $V_{out} = G(V_{in1}, V_{in2}, ..., V_{ink})$. The set of inverter equivalent input assignments to G, denoted $IE(G)$, is a set of k-tuples, $(ie_1, ie_2, ..., ie_k)$, where $ie_i \in (\mathtt{1}, \mathtt{0}, \mathtt{in})$, and $(ie_1, ie_2, ..., ie_k) \in IE(G)$ if and only if $G(ie_1, ie_2, ..., ie_k)$ is functionally equivalent to an inverter with input in, and output out. For the NAND2, the three inverter equivalent input assignments are (1) $(\mathtt{1}, \mathtt{in})$, (2) $(\mathtt{in}, \mathtt{1})$, and (3) $(\mathtt{in}, \mathtt{in})$. In order to work with inverter equivalent input assignments, it is convenient to define F, a simple mapping function between real voltages and elements of $(\mathtt{1}, \mathtt{0}, \mathtt{in})$. That is,

$$F(V_i) = \begin{cases} \mathtt{0} & \text{if } V_i = GND \\ \mathtt{1} & \text{if } V_i = V_{DD} \\ \mathtt{in} & \text{otherwise.} \end{cases} \tag{28}$$

With a notion of inverter equivalence, it is possible to define a representative set of unity gain points for a gate. Let G

be an inverting binary CMOS gate with k inputs labeled as $in1, in2, ..., ink$, and a single output, out. The gradient of V_{out} is defined as

$$\nabla V_{out} = \frac{\partial V_{out}}{\partial V_{in1}}\mathbf{i}_1 + \frac{\partial V_{out}}{\partial V_{in2}}\mathbf{i}_2 + ... + \frac{\partial V_{out}}{\partial V_{ink}}\mathbf{i}_k, \quad (29)$$

where $\mathbf{i}_1$, $\mathbf{i}_2$, ..., $\mathbf{i}_k$ are the corresponding unit vectors. The *representative set of unity gain points* for G, $RS(G)$, is defined such that

$(V_{in1}, V_{in2}, ..., V_{ink}, V_{out}) \in RS(G)$ if and only if
$|\nabla V_{out}(V_{in1}, V_{in2}, ..., V_{ink})| = 1$, and
$(F(V_{in1}), F(V_{in2}), ..., F(V_{ink})) \in IE(G)$, and
for all $x, y \in (1, 2, ..., k)$ (30)
if $F(V_{inx}) = \texttt{in}$ and $F(V_{iny}) = \texttt{in}$ then $V_{inx} = V_{iny}$. (31)

For the NAND2, the representative set of unity gain points are depicted in Figure 24 as light-blue, orange, and red circles with annotated values. The values for each slice $(V_{in1}, V_{in2}, V_{out})$ are (1): (1100,470,1066), (1100,626,42) (2): (445,1100,1063), (630,1100,40), (3): (540,540,1062), (674,674,51). These representative points can be mapped back to simple pairs of the form (V_{in}, V_{out}) by using the inverter equivalent input assignment to remove the references to V_{DD}, GND, and shared inputs. For the NAND2 this *reduced* representative set of unity gain points is (1): (470,1066), (626,42) (2): (445,1063), (630,40), (3): (540,1062), (674,51).

Finally, the VTC parameters can be defined using the reduced set of unity gain points. The usual mapping of unity gain points to VTC parameters can be employed, so for the NAND2, (1): $V_{IL} = 470$, $V_{OH} = 1066$, $V_{IH} = 626$, $V_{OL} = 42$, (2): $V_{IL} = 445$, $V_{OH} = 1063$, $V_{IH} = 630$, $V_{OL} = 40$, and (3): $V_{IL} = 540$, $V_{OH} = 1062$, $V_{IH} = 674$, $V_{OL} = 51$. Statistical analysis is greatly simplified when a single set of representative VTC parameters is chosen, but the parameters—as measured with (1), (2), and (3)—differ. It is clear that V_{OH} and V_{OL} are nearly constant; this is expected (see Section IV-A). The two inverter equivalent input assignments where a single input is tied to V_{DD} ((1) and (2)) are highly symmetric and have only slightly different values for V_{IH} and V_{IL}, respectively. The input assignment wherein both inputs are tied together, (3), does differ significantly in terms of V_{IH} and V_{IL} from (1) and (2).

Consider the measurement of V_{IL} performed by sweeping the input(s) from GND to V_{DD} using (1) as compared to (3). The value of V_{IL} corresponds to the greatest input voltage that still results in the output being pulled-up to a logical-1. The NAND2 contains 2 parallel PFETs with gates connected to $in1$ and $in2$, respectively. With input assignment (1), $in1$ is tied to V_{DD}, causing the corresponding PFET to effectively turn off, *i.e.,* it contributes only sub-threshold leakage current to the pull-up network as $in2$ is swept from GND to V_{DD}. As V_{in2} is increased, the corresponding PFET begins to turn off and the NFETs begin to turn on, thus transitioning the output towards a logic-0; V_{IL} is the input voltage at which this transition occurs. With (3), both inputs are tied together, and the parallel PFETs actively pull up the output node together as the input is swept. The parallel PFETs in (3) continue to actively pull up the output node as the input voltage is increased beyond the V_{IL} from (1). As such, V_{IL} as measured with (3) is greater than V_{IL} as measured with (1). An analogous, but reciprocal explanation can be given for V_{IH}. That is, V_{IH} as measured with (1) is greater than V_{IH} as measured with (3).

In order to provide an upper bound on the robustness, the representative set of VTC parameters should be chosen so as to overestimate the probability of failure of a gate. This corresponds to underestimating both NM_H and NM_L; this, in turn, necessitates underestimating V_{IH} and overestimating V_{IL}.[7] Since V_{OH} and V_{OL} are approximately constant across inverter equivalent input assignment slices, the smallest V_{IH} and the largest V_{IL} should be chosen from the *reduced* representative set of unity gain points. For the NAND2 this corresponds to selecting the VTC parameters from different slices: V_{IH} from (1) and V_{IL} from (3).[8] This somewhat complicates the task of gate characterization, so in this paper the VTC parameters from (1) are chosen as the representative set, despite the fact that this simplifying choice slightly underestimates V_{IL}. In terms of calculating the probability of failure, this simplification has little impact.

Finally, the VTC parameters for an arbitrary gate can be defined. Let G be an inverting binary CMOS gate with m inputs and a single output, out, and let $RS(G)$ be the *reduced representative set of unity gain points* for G. Assume that $RS(G)$ has cardinality n, and elements labeled as (V_{in_i}, V_{out_i}) for $i \in \{1, 2, ..., n\}$. The VTC parameters for G are defined as

$$V_{IH}(G) = min(V_{in_i})$$
$$V_{IL}(G) = max(V_{in_j})$$
$$V_{OH}(G) = min(V_{out_k})$$
$$V_{OL}(G) = max(V_{out_l}),$$

for $i, j, k, l \in \{1, 2, ..., n\}$.

B. Statistical Noise Margins of Combinational Gates

In order to give general definitions for NM_H and NM_L, it is necessary to consider the input VTC parameter correlation between multiple inputs of the same gate. As shown in Figure 26, the input VTC parameters are highly uncorrelated over a wide range of V_{DD}. Given this and the treatment of output VTC parameters as regular variables, arbitrary networks of combinational gates with fan-in and fan-out greater than unity can be broken apart into equivalent gate-pairs, and ultimately, into inverter-equivalent pairs for statistical analysis; *i.e.,* for the purpose of computing the probability of circuit failure.

Consider a circuit, C_a, composed of a network of n combinational gates in an array of simple linear chains; C_a is

[7] Assuming that the corresponding variances are approximately equal.

[8] In a similar fashion, the smallest V_{OH} and largest V_{OL} could be chosen as representative VTC parameters; however, the output VTC parameters are approximately constant, so they can also be chosen arbitrarily or by convenience.

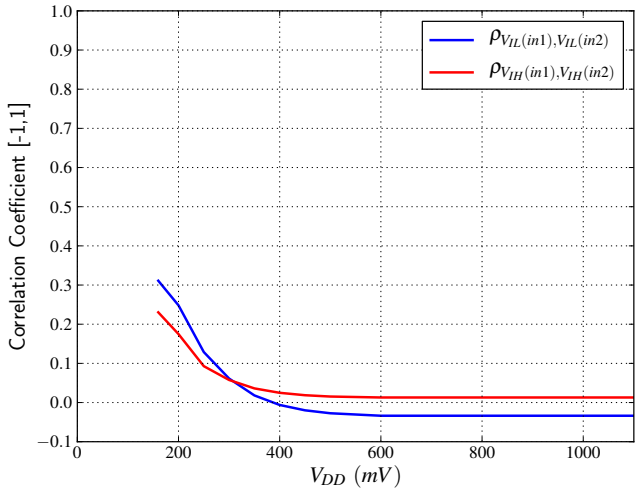

Fig. 26: NAND2 Input Correlation in a commercial 40-nm low-power CMOS process ($V_{DD} = 1.1V$, $25°C$, TT-Corner).

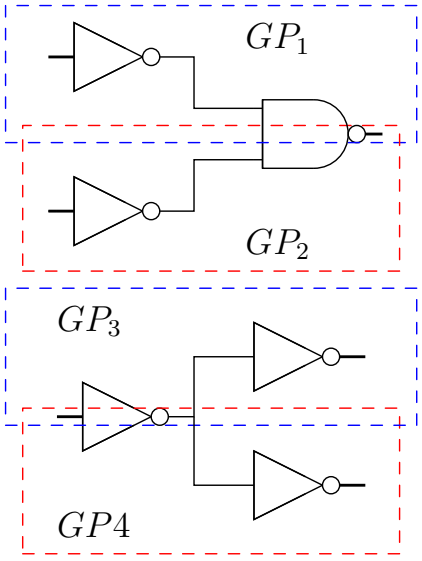

Fig. 27: Equivalent gate-pairs formed from multiple fan-in and fan-out gate networks. GP_1 and GP_2 are formed for each input of the NAND gate, and GP_3 and GP_4 are due to the inverter fan-out.

said to fail if any chain of gates within the circuit fails (see Equation 24). In general, digital circuits consist of networks of combinational gates organized as interconnecting chains, wherein some gates drive multiple gates, some gates are driven by multiple gates, or both. Consider a circuit, C_b, composed of a network of n combinational gates with interconnecting chains, *i.e.,* some gates within C_b drive multiple gates and some gates have multiple inputs. Consider a gate, $G_x \in C_b$ that drives k gates in C_b, labeled as G_1, G_2, ..., G_k. Since the output VTC parameters of G_x are not stochastic in nature, these gates can be treated as k equivalent gate-pairs (G_x, G_i) for $i \in \{1, 2, ..., k\}$. As an example, consider GP_3 and GP_4 in Figure 27. Similarly, consider k gates in C_b, labeled as G_3, G_4, ..., G_k that drive (fan-in) to a multi-input gate, $G_x \in C_b$. Given that the input VTC parameters of G_x are independent, each pair, (G_i, G_x) for $i \in \{1, 2, ..., k\}$, can be considered as components of independent equivalent gate-pairs, as illustrated in 27 by GP_1 and GP_2. Finally, as discussed in Section V-A, every equivalent gate-pair can be analyzed as an inverter equivalent pair, and the robustness of a circuit consisting of arbitrary connections of combinational gates can be computed by way of the methods detailed in Section IV-E. Figure 28 plots the failure probabilities for a linear chain of 20 gates: alternating NAND2 and NOR2 with $NM_T = 10\%V_{DD}$. Similarly, Figure 29 shows the failure probabilities for a chain consisting of alternating NAND3 and NOR3 gates with $NM_T = 20\%V_{DD}$.

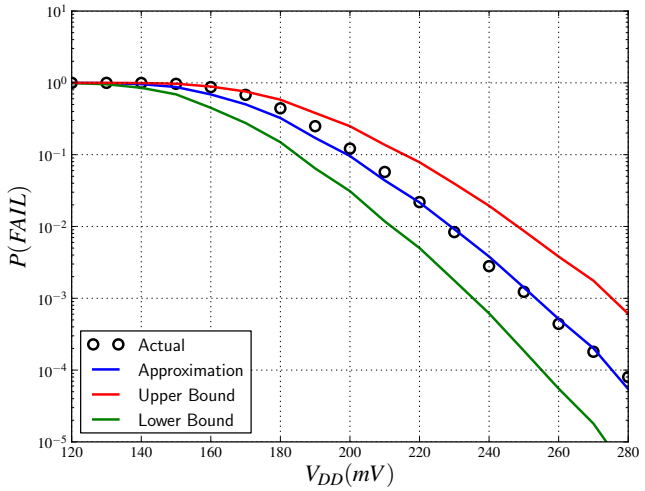

Fig. 28: Probability of chain of 20 combinational gates failing (the chain consists of alternating NAND2, NOR2 gates) with $NM_T = 10\%V_{DD}$ in a commercial 40-nm low-power CMOS process ($25°C$, TT-Corner). For the heuristic approximation, the mean absolute error is 16%, and the maximum absolute error is 36% with $\delta = -1.2\%V_{DD}$.

C. Applications

The methods presented in this paper give a circuit designer the ability to calculate the robustness of a digital circuit composed out of gates. That is, for some circuit, C_a, and a target noise margin, NM_T, Equation 26 gives the probability that some gate chain in C_a has a noise margin less than the target, *i.e.,* a probability of failure $P(FAIL)$. This quantity can, of course, instead be considered as a passing probability $P(PASS)$ by subtracting it from unity, and this passing probability can be thought of as a parametric yield. That is, if $P(PASS) = 95\%$, then in 95% of instances of C_a, all gates will exceed the noise margin target constraint (this is definitionally a parametric yield). If the circuit under consideration,

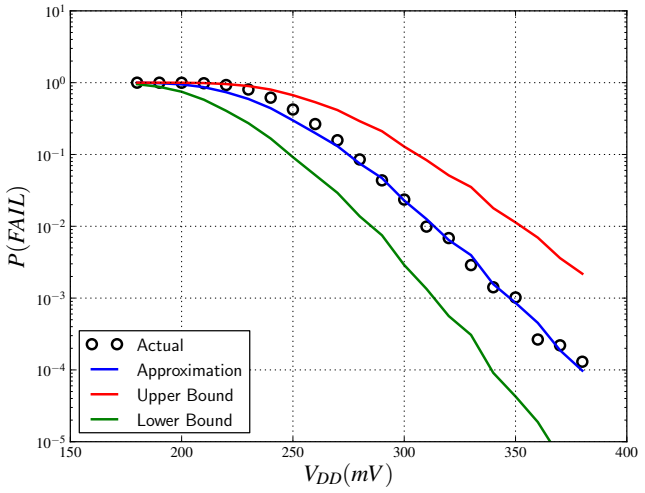

Fig. 29: Probability of chain of 20 combinational gates failing (the chain consists of alternating NAND3, NOR3 gates) with $NM_T = 20\%V_{DD}$ in a commercial 40-nm low-power CMOS process ($25°C$, TT-Corner). For the heuristic approximation, the mean absolute error is 16%, and the maximum absolute error is 67% with $\delta = -1.3\%V_{DD}$.

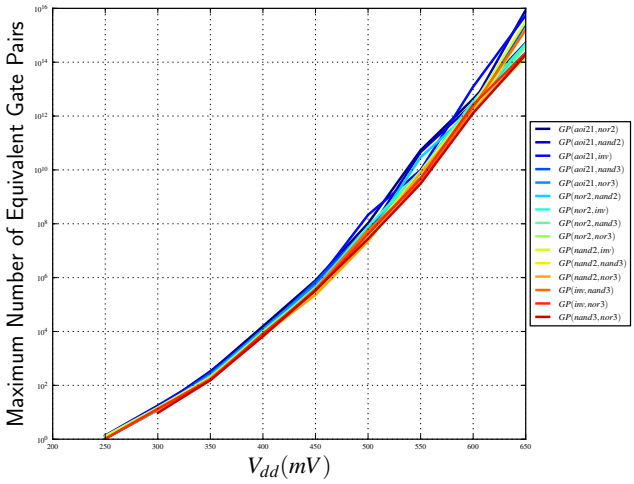

Fig. 30: Maximum number of equivalent gate-pairs vs. V_{DD} with $NM_T = 20\%V_{DD}$ and $yield = 95\%$ in a commercial 40-nm low-power CMOS process ($25°C$, TT-Corner). Chains consist of alternating gates, and all combinations from the set $(INV, NAND2, NOR2, AOI21, NAND3, NOR3)$ are considered.

C_a, is an entire microprocessor, then this parametric yield can be included as a part of the die yield calculation.

In Equation 26, the circuit under consideration, C_a, is an independent variable, and $P(FAIL)$ is a dependent variable. It is straightforward to instead treat C_a as dependent on $P(FAIL)$. In this way, a designer can choose a NM_T and a yield, and then calculate the maximum number of gates that satisfy this constraint (*i.e.,* how large of a circuit can be built). Figure 30 plots the maximum number of equivalent gate-pairs that satisfy a NM_T and yield constraint vs. V_{DD}. It is clear from this figure that the gate choice has only a small impact on how large of circuit can be constructed, and the most important constraint is supply voltage; *i.e.,* the maximum circuit size is exponential in V_{DD}. Figure 31 plots the maximum NM_T that can be guaranteed (for 1M equivalent gate-pairs in chains and a yield of 95%) versus V_{DD}.

VI. Related Work

The earliest works to consider digital circuit robustness with respect to noise and a definition of a static noise margin come independently from Lo and Hill, respectively [32]–[34]. More recently, Shepard proposed a systematic approach to incorporating noise margins into the design process of large circuits via Harmony (an EDA tool) [30]. The primary problem with Harmony is that it does not account for parameter variation, so it is not sufficient for modern low-voltage circuit analysis. It may be possible to apply the robustness metrics and computation techniques detailed in this paper to a tool like Harmony, but this is left as future work.

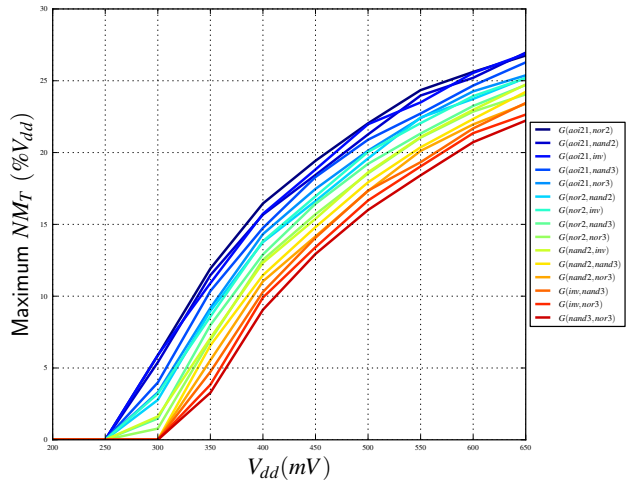

Fig. 31: Maximum NM_T vs. V_{DD} for $1M$ equivalent gate-pairs and $yield = 95\%$ in a commercial 40-nm low-power CMOS process ($25°C$, TT-Corner). Chains consist of alternating gates, and all combinations from the set $(INV, NAND2, NOR2, AOI21, NAND3, NOR3)$ are considered.

Noise margin based analysis of memory cells [39] is common, and a number of works consider the effects of parameter variation in SRAM based analysis, *e.g.,* [4], [11], [12], [36], [40], [42], [43]. These works perform statistical analysis using the SNM expression (Equation 3) and are thus limited by the *min* function. Calhoun works around this somewhat in [36] by using order statistics on the tail of the SNM distribution. A few works (*e.g.,* [10], [13], [44], [45]) analyze combinational gate failures due to parameter variation when operating at low voltages, but these works only consider the single case where all inputs of each gate are tied together. Moreover, these works only consider a simple binary failure model (*i.e.,* $SNM > 0$ or $SNM < 0$), as opposed to the generalized noise margin target based analysis presented in this paper.

Several works specifically consider and model some of the effects of parameter variation on circuits operating subthreshold. Chen considers the limitations in terms of large fan-ins and fan-outs in [9], and Pu [46] uses affine arithmetic to model the effects of parameter variation on V_{OH} and V_{OL}. Alioto derives an accurate closed-form subthreshold SNM model in [14] and considers the effects of variation on subthreshold circuits by way of analyzing the imbalance factor (IF) between the PFET and NFET networks that make up a gate. Some of the work discussed in this paper was initially presented in [16].

VII. Conclusion

This paper presents a metric for digital circuit robustness with respect to parameter variation and noise. The robustness metric is general, and while only applied to CMOS circuits in this paper, can be extended to other technologies (possible future work). Additionally, a compact method for calculating the robustness of CMOS circuits operating sub-threshold or near-threshold is detailed and validated. The method of calculation relies on a new compact representation of parameter variation at the cell level; as such, the robustness of an extremely large circuit can be quickly, efficiently, and accurately computed. The statistical details of the model are flushed out and validated against SPICE simulations of foundry provided statistical BSIM simulations in a modern (40-nm) technology. This work relies extensively on the notion of a static noise margin (see Section III-A). This notion, previously defined exclusively for cross-coupled gate-pairs, is extended in three important ways:

- it is turned into a statistical quantity in Section IV-B,
- it is extended to cover chains of gates in Section IV-E, and
- it is generalized for use with any inverting single-output CMOS gate in Section V-A.

As with all metrics, there are limitations to the applicability of the work presented in this paper. Many of the calculations rely on the assumption of statistical independence of parameter variation between different gates. This assumption is discussed in detail and justified in Section IV-B. If this assumption does not hold, the effects of correlation can be accounted for by way of adding a correlation coefficient to Equations 16, 20, and 26. These effects are likely well modeled as spatial correlation [47], so accurate correlation models may require knowledge of circuit layout. Quantifying these effects, which are currently only significant at high supply voltages, is left as future work. Finally, choosing different noise margin targets for different gates is left as future work (circuit noise can vary from gate to gate).

References

[1] S. Borkar, N. Jouppi, and P. Stenstrom, "Microprocessors in the era of terascale integration," in *Design, Automation Test in Europe Conference Exhibition, 2007. DATE '07*, 2007, pp. 1–6.

[2] N. H. E. Weste and D. M. Harris, *CMOS VLSI Design, 4th Ed.* Addison-Wesley, 2010.

[3] C. Hawkins, J. Soden, A. Righter, and F. Ferguson, "Defect classes-an overdue paradigm for CMOS IC testing," in *Test Conference, 1994. Proceedings., International*, 1994, pp. 413–425.

[4] S. Mukhopadhyay, H. Mahmoodi, and K. Roy, "Modeling of failure probability and statistical design of SRAM array for yield enhancement in nanoscaled CMOS," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 24, no. 12, pp. 1859–1880, 2005.

[5] *International Technology Roadmap for Semiconductors*, 2011.

[6] M. Alioto, "Ultra-low power vlsi circuit design demystified and explained: A tutorial," *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 59, no. 1, pp. 3 –29, Jan. 2012.

[7] J. Chen, L. Clark, and Y. Cao, "Robust design of high fan-in/out subthreshold circuits," in *Computer Design: VLSI in Computers and Processors, 2005. ICCD 2005. Proceedings. 2005 IEEE International Conference on*, Oct. 2005, pp. 405 – 410.

[8] B. Calhoun, A. Wang, and A. Chandrakasan, "Modeling and sizing for minimum energy operation in subthreshold circuits," *Solid-State Circuits, IEEE Journal of*, vol. 40, no. 9, pp. 1778 – 1786, Sep. 2005.

[9] J. Chen, L. Clark, and Y. Cao, "Maximum - ultra-low voltage circuit design in the presence of variations," *Circuits and Devices Magazine, IEEE*, vol. 21, no. 6, pp. 12 – 20, 2005.

[10] J. Kwong and A. Chandrakasan, "Variation-driven device sizing for minimum energy sub-threshold circuits," in *Low Power Electronics and Design, 2006. ISLPED'06. Proceedings of the 2006 International Symposium on*, Oct. 2006, pp. 8 –13.

[11] E. Grossar, M. Stucchi, K. Maex, and W. Dehaene, "Read stability and write-ability analysis of sram cells for nanometer technologies," *Solid-State Circuits, IEEE Journal of*, vol. 41, no. 11, pp. 2577 –2588, Nov. 2006.

[12] N. Verma, J. Kwong, and A. Chandrakasan, "Nanometer MOSFET variation in minimum energy subthreshold circuits," *Electron Devices, IEEE Transactions on*, vol. 55, no. 1, pp. 163 –174, Jan. 2008.

[13] D. Bol, R. Ambroise, D. Flandre, and J.-D. Legat, "Interests and limitations of technology scaling for subthreshold logic," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, vol. 17, no. 10, pp. 1508 –1519, Oct. 2009.

[14] M. Alioto, "Understanding DC behavior of subthreshold CMOS logic through closed-form analysis," *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 57, no. 7, pp. 1597 –1607, Jul. 2010.

[15] D. Harris, B. Keller, J. Karl, and S. Keller, "A transregional model for near-threshold circuits with application to minimum-energy operation," in *Microelectronics (ICM), 2010 International Conference on*, Dec. 2010, pp. 64 –67.

[16] S. Keller, S. S. Bhargav, C. Moore, and A. J. Martin, "Reliable minimum energy CMOS circuit design," in *2nd European Workshop on CMOS Variability (VARI 2011)*, Grenoble, France, May 2011.

[17] B. Zhai, D. Blaauw, D. Sylvester, and K. Flautner, "Theoretical and practical limits of dynamic voltage scaling," in *Design Automation Conference, 2004. Proceedings. 41st*, 2004, pp. 868 – 873.

[18] D. Markovic, C. Wang, L. Alarcon, T.-T. Liu, and J. Rabaey, "Ultralow-power design in near-threshold region," *Proceedings of the IEEE*, vol. 98, no. 2, pp. 237 –252, Feb. 2010.

[19] R. Dreslinski, M. Wieckowski, D. Blaauw, D. Sylvester, and T. Mudge, "Near-threshold computing: Reclaiming moore's law through energy efficient integrated circuits," *Proceedings of the IEEE*, vol. 98, no. 2, pp. 253 –266, Feb. 2010.

[20] L. Chang, D. Frank, R. Montoye, S. Koester, B. Ji, P. Coteus, R. Dennard, and W. Haensch, "Practical strategies for power-efficient computing technologies," *Proceedings of the IEEE*, vol. 98, no. 2, pp. 215 –236, Feb. 2010.

[21] K. Shepard and V. Narayanan, "Conquering noise in deep-submicron digital ICs," *Design Test of Computers, IEEE*, vol. 15, no. 1, pp. 51 –62, Jan.-Mar. 1998.

[22] K. Bernstein, D. J. Frank, A. E. Gattiker, W. Haensch, B. L. Ji, S. R. Nassif, E. J. Nowak, D. J. Pearson, and N. J. Rohrer, "High-performance CMOS variability in the 65-nm regime and beyond," *IBM Journal of Research and Development*, vol. 50, no. 4.5, pp. 433 –449, Jul. 2006.

[23] P. Asenov, N. Kamsani, D. Reid, C. Millar, S. Roy, and A. Asenov, "Combining process and statistical variability in the evaluation of the effectiveness of corners in digital circuit parametric yield analysis," in *Solid-State Device Research Conference (ESSDERC), 2010 Proceedings of the European*, 2010, pp. 130–133.

[24] K. Okada and N. Onodera, "Statistical modeling of device characteristics with systematic fluctuation," in *Circuits and Systems, 2000. Proceedings. ISCAS 2000 Geneva. The 2000 IEEE International Symposium on*, vol. 2, 2000, pp. 437–440 vol.2.

[25] B. Cheng, S. Roy, and A. Asenov, "Impact of intrinsic parameter fluctuations on deca-nanometer circuits, and circuit modelling techniques," in *Mixed Design of Integrated Circuits and System, 2006. MIXDES 2006. Proceedings of the International Conference*, 2006, pp. 117–121.

[26] N. Drego, A. Chandrakasan, and D. Boning, "Lack of spatial correlation in mosfet threshold voltage variation and implications for voltage scaling," *Semiconductor Manufacturing, IEEE Transactions on*, vol. 22, no. 2, pp. 245–255, 2009.

[27] P. Friedberg, Y. Cao, J. Cain, R. Wang, J. Rabaey, and C. Spanos, "Modeling within-die spatial correlation effects for process-design co-optimization," in *Quality of Electronic Design, 2005. ISQED 2005. Sixth International Symposium on*, 2005, pp. 516–521.

[28] P. Asenov, F. Adamu-Lema, S. Roy, C. Millar, A. Asenov, G. Roy, U. Kovac, and D. Reid, "The effect of compact modelling strategy on SNM and read current variability in modern SRAM," in *Simulation of Semiconductor Processes and Devices (SISPAD), 2011 International Conference on*, 2011, pp. 283–286.

[29] A. Scholten, L. Tiemeijer, R. Van Langevelde, R. Havens, A. Zegers-van Duijnhoven, and V. Venezia, "Noise modeling for RF CMOS circuit simulation," *Electron Devices, IEEE Transactions on*, vol. 50, no. 3, pp. 618–632, 2003.

[30] K. Shepard, V. Narayanan, and R. Rose, "Harmony: static noise analysis of deep submicron digital integrated circuits," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 18, no. 8, pp. 1132 –1150, Aug. 1999.

[31] L. Ding and P. Mazumder, "Dynamic noise margin: definitions and model," in *VLSI Design, 2004. Proceedings. 17th International Conference on*, 2004, pp. 1001–1006.

[32] A. Lo, "Physical realization of digital logic circuits," in *Micropower Electronics*. New York, NY: Macmillan, 1964, pp. 19–39.

[33] C. F. Hill, "Definitions of noise marign in logic systems," *Mullard Technical Communications*, vol. 89, pp. 239–245, Sep. 1967.

[34] ——, "Noise margin and noise immunity of logic circuits," *Microelectronics*, vol. 1, pp. 16– 21, Apr. 1968.

[35] J. Meindl and J. Davis, "The fundamental limit on binary switching energy for terascale integration (TSI)," *Solid-State Circuits, IEEE Journal of*, vol. 35, no. 10, pp. 1515 –1516, Oct. 2000.

[36] B. Calhoun and A. Chandrakasan, "Static noise margin variation for subthreshold SRAM in 65-nm CMOS," *Solid-State Circuits, IEEE Journal of*, vol. 41, no. 7, pp. 1673 –1679, Jul. 2006.

[37] J. Lohstroh, E. Seevinck, and J. de Groot, "Worst-case static noise margin criteria for logic circuits and their mathematical equivalence," *Solid-State Circuits, IEEE Journal of*, vol. 18, no. 6, pp. 803–807, 1983.

[38] J. R. Hauser, "Noise margin criteria for digital logic circuits," *Education, IEEE Transactions on*, vol. 36, no. 4, pp. 363–368, 1993.

[39] E. Seevinck, F. List, and J. Lohstroh, "Static-noise margin analysis of MOS SRAM cells," *Solid-State Circuits, IEEE Journal of*, vol. 22, no. 5, pp. 748–754, 1987.

[40] A. J. Bhavnagarwala, X. Tang, and J. D. Meindl, "The impact of intrinsic device fluctuations on CMOS SRAM cell stability," *Solid-State Circuits, IEEE Journal of*, vol. 36, no. 4, pp. 658–665, 2001.

[41] P. Heydari and M. Pedram, "Ground bounce in digital vlsi circuits," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, vol. 11, no. 2, pp. 180–193, 2003.

[42] B. Calhoun and A. Chandrakasan, "Analyzing static noise margin for sub-threshold SRAM in 65nm CMOS," in *Solid-State Circuits Conference, 2005. ESSCIRC 2005. Proceedings of the 31st European*, 2005, pp. 363–366.

[43] M. Seok, D. Sylvester, and D. Blaauw, "Optimal technology selection for minimizing energy and variability in low voltage applications," in *Low Power Electronics and Design (ISLPED), 2008 ACM/IEEE International Symposium on*, 2008, pp. 9–14.

[44] D. Bol, R. Ambroise, D. Flandre, and J. Legat, "Analysis and minimization of practical energy in 45nm subthreshold logic circuits," in *Computer Design, 2008. ICCD 2008. IEEE International Conference on*, 2008, pp. 294–300.

[45] M. Korbel, D. Stow, C. Ferguson, and D. Harris, "Yield-driven minimum energy CMOS cell design," in *Signals, Systems and Computers (ASILOMAR), 2012 Conference Record of the Forty Sixth Asilomar Conference on*, 2012, pp. 1010–1014.

[46] Y. Pu, J. de Gyvez, H. Corporaal, and Y. Ha, "Statistical noise margin estimation for sub-threshold combinational circuits," in *Design Automation Conference, 2008. ASPDAC 2008. Asia and South Pacific*, 2008, pp. 176–179.

[47] D. Sylvester, K. Agarwal, and S. Shah, "Variability in nanometer CMOS: Impact, analysis, and minimization," *Integration, the VLSI Journal*, vol. 41, no. 3, pp. 319–339, 2008.

[48] A. Singhee and R. Rutenbar, "Why quasi-Monte Carlo is better than Monte Carlo or latin hypercube sampling for statistical circuit analysis," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on*, vol. 29, no. 11, pp. 1763 –1776, Nov. 2010.

[49] B. Cheng, S. Roy, and A. Asenov, "Impact of intrinsic parameter fluctuations on deca-nanometer circuits, and circuit modelling techniques," in *Mixed Design of Integrated Circuits and System, 2006. MIXDES 2006. Proceedings of the International Conference*, Jun. 2006, pp. 117 –121.

[50] B. Sheu, D. Scharfetter, P.-K. Ko, and M.-C. Jeng, "BSIM: Berkeley short-channel IGFET model for MOS transistors," *Solid-State Circuits, IEEE Journal of*, vol. 22, no. 4, pp. 558 – 566, Aug. 1987.